Dissertation
submitted to the
Combined Faculties for the Natural Sciences and for Mathematics
of the Ruperto-Carola University of Heidelberg, Germany
for the degree of
Doctor of Natural Sciences

put forward by
Dipl.-Phys. Julian Stapf
born in Bruchsal

Date of oral exam: January, 21th 2015

Novel learning-based techniques for dense fluid motion measurements

Referees:

Priv.-Doz. Dr. Christoph S. Garbe

Prof. Dr. Werner Aeschbach-Hertig

Abstract: In this thesis novel learning-based approaches are presented for the estimation of dense fluid flow velocity fields from particle image sequences. The developed methods apply prior knowledge in form of typical spatio-temporal motion models. These motion models are obtained with methods of unsupervised learning using *proper orthogonal decomposition* (POD). The POD modes reveal dominant flow structures and contain relevant information of complex relations between neighboring flow vectors. The first high-energy POD modes obtained from appropriate training vector fields are used as typical motion models. Meaningful local flow structures can be expressed in the orthogonal space spanned by the motion models. Additional information about dominant flow events is gained by the motion models and related parameters. The proposed approaches are embedded in well-established local parametric and variational optical flow frameworks but are contrasted with these common techniques by the inclusion of prior knowledge. Further extensions of the methods use available information, which is generally discarded in other methods, to obtain robust motion estimations. The methods can easily be tuned for different flow applications by choice of training data and, thus, are universally applicable. Beyond their simple implementation, the approaches are very efficient, accurate and easily adaptable to all types of flow situations. All methods were tested on synthetic and real particle image sequences and the influence of the relevant parameters was investigated. For typical use cases of optical flow, such as small image displacements, they were more accurate compared to all competing methods including *particle image velocimetry* (PIV) and common optical flow techniques.

Zusammenfassung: In dieser Arbeit werden neue lernbasierte Methoden zur Bestimmung dichter Strömungsgeschwindigkeitsfelder aus Tracer-Partikel-Bildsequenzen vorgestellt. Dabei wird Vorwissen in Form von typischen räumlich-zeitlichen Bewegungsmustern verwendet. Diese Bewegungsmuster werden durch Methoden des *unsupervised learnings* mittels Hauptkomponentenzerlegung (POD) bestimmt. Die POD-Moden beschreiben dominante Strömungsstrukturen und beinhalten Informationen über den komplexen Zusammenhang benachbarter Geschwindigkeitsvektoren. Die ersten, aus geeigneten Trainingsvektorfeldern gelernten, POD-Moden mit hohem Informationsgehalt bilden die typischen Bewegungsmuster. Sinnvolle Strömungsstrukturen liegen im Lösungsraum, der von den Bewegungsmustern aufgespannt wird. Aus den Bewegungsmustern kann, zusammen mit entsprechenden Parametern, Zusatzinformation über vorherrschende Strömungsmuster gewonnen werden. Die vorgeschlagenen Ansätze basieren auf gängigen *optischer Fluss* Schätzern, die zusätzlich das Vorwissen aus den gelernten Bewegungsmustern einbeziehen. Um die Ergebnisse robuster zu machen, wird bereits vorhandene Information verwendet, die von bisherigen Schätzern unberücksichtigt geblieben ist. Durch Wahl geeigneter Trainingsdaten können die Methoden leicht für verschiedene Anwendungen optimiert werden und sind dadurch universell einsetzbar. Darüber hinaus sind sie einfach zu implementieren, effizient, genau und anpassungsfähig. Alle Varianten wurden an synthetischen und realen Sequenzen getestet und der Einfluss der relevanten Parameter wurde untersucht. Für kleine Verschiebungen zwischen einzelnen Bildern erzielten die lernbasierten Methoden verglichen mit anderen getesteten Methoden, wie z. B. *particle image velocimetry* (PIV) oder bekannten *optischer Fluss* Schätzern, die genausten Ergebnisse.

Contents

1 Introduction

1.1 Motivation

In numerous technical applications a precise measurement of fluid flow is essential. Reduction of energy usage or fluctuating forces in aerospace applications, combustion processes or technical flows can often only be achieved through experimental measurements of the fluid flow. Also for the validation and calibration of numerical models precise measurements of the flow field are required. Of particular interest in many technical processes is the flow in near-wall regions as well as in turbulent areas. For instance in combustion processes, turbulence is exploited in order to mix the fuel charge and increase the combustion speed. However, the amount of turbulence is critical, since strong turbulence leads to noise and efficiency losses. Considering airfoils turbulence puts a lot of stress on certain parts and reduces their durability. The boundary layer flow around objects must be investigated in order to reduce efficiency losses due to drag. For an optimization of these processes an accurate measurement of the fluid velocity field is required. Due to many fluctuating small scale structures contained in turbulent flows, a dense instantaneous determination of the flow field is essential. The development and improvement of capable measurement techniques, which fulfill this requirement, is the aim of this thesis.

The motion field is usually determined using flow visualization and recording techniques, which have the advantage that they are *non-intrusive* and are able to capture instantaneous velocity fields. Accordingly, tracer material, such as dye or neutrally buoyant particles, is added to the fluid and the now visible flow is recorded with a high-speed camera. The motion field is estimated from the recorded image sequence measuring the displacements of corresponding particles in subsequent images. If the time between consecutive frames is considered, the velocity can be deduced from the displacement. In order to estimate the displacement field, usually correlation-based methods, denoted by *particle image velocimetry* (PIV) are applied. The displacement is thereby given by the maximum value of the cross-correlation coefficient of two corresponding local areas taken from subsequent images. However, correlation-based PIV suffers from some limitations such as a low spatial resolution, a constriction to integer values, and a missing integration of the underlying physics. Most of these limitations can be tackled to a certain extend by additional assumptions and post-processing steps, but the accurate estimation of flow fields on small scales remains challenging.

There are also other methods to estimate motion from image sequences originating from computer vision. The technique of *optical flow* was originally developed to estimate the movement

of the recording camera as well as the motion of imaged objects. Usually, it relies on the assumption of conserved image brightness. Especially *variational* methods have the advantage to yield dense flow fields providing one flow estimate per pixel. Although optical flow was originally established without particular awareness of fluid mechanical problems, the application for the measurement of fluid motion seems natural. Especially, in turbulent regions optical flow methods are reasonable due to their dense description of the flow field.

In this thesis new concepts and approaches for dense, accurate fluid motion estimation are presented. These methods extent the established optical flow techniques by using prior knowledge of meaningful flow structures represented by previously learned typical motion models.

1.2 Previous work

The systematic, experimental investigation of fluid flows started in the early 20th century. Prandtl [1904] used flow visualization techniques to study unsteady, separated flows behind objects in a water tunnel [Merzkirch, 2007]. In order to make the flow visible, he added a suspension of mica particles to the water surface [Raffel et al., 2007]. With these qualitative techniques only the geometry and the orientation of flow structures could be studied, but a thorough, quantitative evaluation of the motion field was not possible. In order to determine the fluid velocity, single-point measurements (e.g., with a pitot tube), could be conducted [Adrian and Westerweel, 2011]. For many interesting flow features such as the vorticity, the knowledge of the instantaneous velocity field is required. Therefore, techniques were needed, which are able to measure the complete flow field at one instant of time. This led among other methods to the development of whole field techniques, with which the velocity field is determined from particle image sequences taken from the fluid flow, such as correlation-based PIV [Adrian, 1991; Willert and Gharib, 1991]. Whereas it was an analog technique at the beginning, especially the technical progress made in optics, lasers, electronics, video and computer techniques helped to establish a greatly improved digital version. To date, many improvements and extensions to PIV have been proposed [Adrian and Westerweel, 2011; Raffel et al., 2007] and led to a wide usage and popularity of the method.

Almost 20 years ago the potential of optical flow techniques in order to estimate fluid motion was recognized [Quénot et al., 1998; Wildes et al., 1997]. This technique was originally developed in computer vision some time before correlation-based PIV [Horn and Schunck, 1981; Lucas and Kanade, 1981]. The concept is widely used in object detection and tracking, robot navigation, control systems, as well as video compression. In their seminal work Horn and Schunck [1981] solved the optical flow problem in a variational manner by minimizing an energy functional consisting of a data term and an additional regularization term. Their approach was the blueprint for many other techniques to come. Lucas and Kanade [1981] took a different approach by assuming constant flow in local neighborhoods pooling different constraints derived within the neighborhood for a single joint estimate.

Since optical flow was developed for the motion estimation of large, rigid objects, it had to be slightly adapted to meet the requirements of fluid motion detection. The regularization term proposed by Horn and Schunck [1981] suppresses the divergence and the vorticity of the flow field. Because this suppression is unwanted in most fluid flows, Corpetti et al. [2002] proposed to use divergence and vorticity preserving regularizers introduced by Suter [1994]. In many applications the assumption of a constant image brightness is violated. As a consequence Brox et al. [2004] assumed other image features such as the gradient of the brightness to be constant. Haussecker and Fleet [2001] used a parameterized model to describe brightness changes along streamlines. Physically more grounded data terms that build on the *continuity equation* were considered to be a good choice for fluid motion estimation [Corpetti et al., 2006; Liu and Shen, 2008]. According to Heitz et al. [2010], variational optical flow methods provide an adequate framework in order to combine image measurements with physical constraints derived from the fluid. Such methods establish a proper connection between computational fluid dynamics and measurement data. Nakajima et al. [2003] proposed to use the *Navier-Stokes equations* as regularization term. This method prefers solutions that follow the underlying physical equations. A further approach combining image-based flow measurements with prior physical knowledge in form of the constitutive fluid flow equations was proposed for viscose non-turbulent flows by Ruhnau and Schnörr [2007]. This approach was also extended to turbulent flows using the 2D vorticity transport equation [Ruhnau et al., 2007]. An overview of the basic fluid motion estimation schemes presented in the past 20 years is given in Heitz et al. [2010].

Optical flow approaches using previous knowledge learned from appropriate training data were proposed by Black et al. [1997] and Yacoob and Davis [1998]. They employed spatial or temporal motion models to estimate non-rigid flow fields of human action like walking or speaking. These approaches were refined by Nieuwenhuis et al. [2010] to fit into a simple parametric optical flow framework with spatio-temporal motion models. The approaches proposed in this thesis extend these learning-based techniques in order to estimate dense motion fields of fluid flows. The methods are able to use prior knowledge about complex local flow structures in a simple but efficient way. This is in contrast to other optical flow methods that apply prior knowledge such as the method proposed by Ruhnau et al. [2007]. These physics-based methods are extremely complex and extensive and require the complete knowledge of the boundary conditions, which is often not feasible. The proposed approaches are particularly of interest, if prior knowledge is available in form of similar flow fields or numerical simulations, but the boundary conditions are unknown.

1.3 Scope of this thesis

The scope of this thesis is to develop new, improved estimation methods in order to determine dense velocity fields of fluid flows based on particle images. Therefore, previous knowledge about local flow structures in form of typical motion models is applied and the estimated flow

fields are expressed in the solution space spanned by these orthogonal basis flows. The motion models are obtained by methods of unsupervised learning. Accordingly, flow estimates fulfill the guidelines defined by the motion models and are more accurate than the estimates from standard techniques. Essentially, the typical motion models are embodied by the high energy modes obtained from a *proper orthogonal decomposition* (POD). The proposed approaches incorporate the experience and know-how of over thirty years of research in the field of computer vision and are integrated into existing well-established gradient-based optical flow frameworks. The method is either realized as purely local parametric approach or as combined local global variational technique. By the application of a cost function, which can easily be adapted to model the underlying effects, the approach is very flexible. Advantages of these methods such as the achievement of dense high resolution flow fields are also given for the proposed methods. A further extension of these approaches uses information obtained from neighboring estimates, which is usually discarded by the existing methods, in a statistical manner to yield results of increased accuracy. The benefits and limitations of the different approaches are investigated and compared to each other and the properties and conditions yielding the best results are examined. The proposed approaches are compared on different fluid dynamical test cases to common optical flow and correlation-based methods in order to rank the quality of the achieved flow fields.

1.4 Structure of the thesis

The remainder of this thesis is organized as follows. In Chapter 2, fundamental fluid dynamical equations and properties are introduced and the mathematical tools and concepts, which are important for an understanding of this thesis, are described. In Chapter 3, the general principles of fluid motion estimation techniques are presented, and common estimation approaches are introduced. Apart from PIV which was especially developed with fluid dynamical applications in mind, also gradient-based optical flow techniques originating from computer vision and adapted for fluid flow measurements are described. In Chapter 4, the approaches developed in this thesis are introduced and the influence of different approach specific properties are investigated on different test sequences. In Chapter 5, tests of the developed approaches on various fluid dynamical test cases are described and the achieved results are compared to the results of other common optical flow and PIV methods. Finally, the general findings of this thesis are summarized and discussed in Chapter 6. All of the abbreviations used within this thesis are listed on page 121.

2 Basics

In this chapter, some basic principles and concepts, which are considered to be helpful in order to understand this thesis, are introduced, starting with a short recapitulation of the fundamental fluid dynamical equations in Section 2.1. In Section 2.2 the POD is introduced, which is an important analysis tool used in many fluid dynamical applications, and which is the centerpiece of the estimation algorithm proposed within this thesis. Finally, in Section 2.3 some basic mathematical concepts, which are needed in order to solve the occurring equations, are presented.

2.1 Fluid dynamics

The field of fluid dynamics describes basically the motion of fluids. In the following section, the important equations are shortly introduced. They are derived from the conservation laws of *mass*, *momentum*, and *energy*. A detailed description of the topic can be found in any fluid mechanical text book, e.g., [Kundu et al., 2011; Spurk and Nuri, 2008].

2.1.1 Continuity equation

The conservation of mass leads to the continuity equation of fluid mechanics, which simply states that the fluid mass entering a volume equals the fluid mass leaving the volume. This means that mass is neither created nor destroyed. In differential form the continuity equation is given by

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{u}) = 0 \tag{2.1}$$

with density ρ and velocity $\boldsymbol{u}$. Using the *material derivative* $\frac{\mathrm{D}\rho}{\mathrm{D}t} := \frac{\partial \rho}{\partial t} + (\boldsymbol{u} \cdot \nabla)\,\rho$, Equation (2.1) can be formulated as

$$\frac{\mathrm{D}\rho}{\mathrm{D}t} + \rho\,(\nabla \cdot \boldsymbol{u}) = 0\,. \tag{2.2}$$

The material derivative describes the time rate of change of the density. It states that the change of ρ is either due to local temporal changes $\frac{\partial \rho}{\partial t}$ or due to advection by the mean flow $\boldsymbol{u}$. Equation (2.2) is given in *Eulerian* description, which means that the characteristics of the flow field are monitored at fixed locations. Also common is the *Lagrangian* description, where fluid particles are followed as they move through the flow field. Depending on the situation, each of the descriptions has its advantages. Within this thesis fluid velocity fields are observed, which corresponds to an Eulerian description.

For *incompressible* fluids, which implies that the density is considered to be constant within a small parcel of fluid, the derivative of the density vanishes ($\frac{\mathrm{D}\rho}{\mathrm{D}t} = 0$). From Equation (2.2) it follows

$$\nabla \boldsymbol{u} = 0 \,. \tag{2.3}$$

Usually, liquids are considered as incompressible fluids. For velocities smaller than approximately $100\,\frac{m}{s}$ (Mach numbers < 0.3), also gases can be considered to be incompressible [Kundu et al., 2011].

2.1.2 Navier-Stokes equation

The *Navier-Stokes* equation describes the motion of Newtonian fluids such as water or air, which can be characterized by a linear proportionality of shear stress and shear rate of the fluid. The solution of the Navier-Stokes equation is a velocity field, also called flow field, which is a description of the velocity of the fluid at a given point in space and time. The Navier-Stokes equation corresponds to *Newtons second law of motion*, which states that the acceleration equals the net forces on an object divided by its mass. The forces acting on a fluid are pressure forces, viscous forces, as well as external forces such as gravity. The acceleration of a fluid parcel can be described by the material derivative $\frac{D\boldsymbol{u}}{Dt} := \frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\,\boldsymbol{u}$. It consists of two parts. The first part describes the rate of change of the velocity at a particular location with respect to time. The second part describes the velocity change due to the transport of the fluid parcel to a location with different velocity.

For incompressible fluids, which fulfill Equation (2.3), the Navier-Stokes equation can be written in the form

$$\rho \left(\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\,\boldsymbol{u} \right) = -\nabla p + \mu \Delta \boldsymbol{u} + \boldsymbol{f} \tag{2.4}$$

with velocity $\boldsymbol{u}$, density ρ, pressure p, dynamic viscosity μ, and body forces per unit volume $\boldsymbol{f}$. The Navier-Stokes equation is a non-linear partial differential equation of second order. The non-linearity is due to the term $(\boldsymbol{u} \cdot \nabla)\,\boldsymbol{u}$ which is quadratic in $\boldsymbol{u}$. Therefore, the solution of Equation (2.4) is very difficult and often only possible under certain assumptions.

For small velocities, the non-linear term is usually neglected and the equation describes laminar flows. However, when the flow turns turbulent, the influence of the non-linear term is strong and cannot be neglected anymore.

2.1.3 Turbulent flows

A fluid flow becomes turbulent, if the so-called *Reynolds number* defined as

$$Re := \frac{\rho U L}{\mu} \tag{2.5}$$

exceeds a problem dependent critical value. Thereby, U denotes the velocity scale and L a characteristic length scale, which depends on the underlying geometry. The Reynolds number itself is dimensionless. The change from laminar to turbulent flows is not sharp. In the interval between low (laminar) and high (turbulent) Reynolds numbers, where both flow types are possible, the flow is called transitional. According to Kundu et al. [2011] turbulence can be imagined as a dissipative flow state characterized by non-linear fluctuating 3D vorticity.

A turbulent flow consists of rotational structures, so-called eddies, of different length scales. *Kolmogorov's* theory describes how energy is transferred from eddies of larger scales to eddies of smaller scales. This energy transfer goes under the name *energy cascade*. Only at the dissipative subrange, at the bottom of the cascade, kinetic energy dissipates into heat. The energy transfer across different length scales is described by the energy spectrum, which is defined as the Fourier transform of the autocorrelation function of the velocity components. According to Kolmogorov's theory, the spectrum decreases with increasing wave number k proportional to $k^{-5/3}$ [Kundu et al., 2011].

However, this behavior is different for 2D turbulent flows. Here, two different inertial ranges are observed [Boffetta et al., 2005]. On the one hand, an inverse Kolmogorov spectrum proportional to $k^{-5/3}$ is present and, thus, energy is transferred from the point of energy injection to larger scales. On the other hand a downscale enstrophy (mean square vorticity) cascade to smaller scales proportional to k^{-3} is present.

A mathematical tool which is often used to study turbulent flows is POD. In this thesis it is used to generate an orthogonal system of basis functions, which serves as solution space for the fluid flow problem. Therefore, it is introduced in the next section.

2.2 Proper orthogonal decomposition

The *proper orthogonal decomposition* (POD) is a powerful mathematical method for data analysis and simplification. It transforms a multi-dimensional dataset of possibly correlated variables into a set of uncorrelated variables, and a new orthogonal system of basis vectors is found, in which the data can be expressed in an optimal manner. The basis found by POD is the most preferred one to use in many applications. Depending on the field of application the POD has many different names such as *Karhunen-Loève decomposition*, *principal component analysis*, or *Hotelling transform*.

Other than in fluid dynamics, POD is used according to Holmes et al. [1996] mainly in the fields of image processing, signal analysis, random variables, and data compression. To compress a large dataset, POD removes redundant information. The dimensionality of the dataset is reduced and the information is redistributed onto new variables that optimally represent the data. Formerly interrelated variables are merged to obtain fewer variables that are unrelated. POD is also used for the reduction of noise in datasets which is also done by a decrease of the dimensionality. This is due to the fact that POD sorts the new optimal variables according to

their variance. Assuming that the proportionate noise variance is less than the total variance of the dataset, the last variables can be cut of.

Mathematically speaking, the POD is equal to a diagonalization of the covariance matrix. In this way the redundancy between the different variables is dropped since all cross variances of the variables become zero. Basically, it is a coordinate transform into a new, simplified coordinate system.

2.2.1 Proper orthogonal decomposition in fluid dynamics

The POD was introduced to the field of fluid dynamics in the context of turbulence by Lumley [1967]. It can contribute to a better understanding of turbulent flows since it is a useful technique to analyze complex flow phenomena and is used for the identification of the most energetic contributions and dominant structures. Therefore, it is sometimes utilized for the discovery and the identification of *coherent structures* as shown by Berkooz et al. [1993] and Sirovich [1987]. In the study of turbulent flows it helps to break the complex, multi-scaled, random fields of turbulent motion down into more elementary organized motions, which are called coherent structures. However, the concept of coherent structures is often vague and lacks a clear definition. Coherent structures are organized spatial features, which repeatedly appear and undergo a characteristic temporal life cycle [Berkooz et al., 1993]. Since the continuity of fluid motion guarantees that any continuous fluid motion is spatially coherent, only motion structures that contribute to time averaged flow statistics are counted as coherent structures [Adrian, 2007]. They can be imagined as larger fluid portions that stick together while moving through the fluid. Coherent structures can also be viewed as energetically dominant recurrent patterns [Haller and Yuan, 2000]. Interesting features of coherent structures are kinematic properties such as size, shape, vorticity, and energy as well as dynamic properties such as stability, origin, growth, and transformation. Coherent structures occur in different scales. In turbulent boundary layers for example the bulges are large-scale coherent structures, and the near-wall quasi-streamwise vortices are small scale structures [Nobach et al., 2007].

The POD yields a complete set of orthogonal basis functions consisting of the eigenfunctions of the correlation matrix of the dataset, which are often called POD modes. It is generally recognized that the empirical eigenfunctions extracted by POD are intimately related to coherent structures, although, the exact relationship is debated [Gordeyev and Thomas, 2000]. Lumley [1981] stated that the first POD mode represents a coherent structure only if it contains a dominant part of the variance. Therefore, sometimes a summation of the most energetic POD modes is considered as the large-scale coherent structure [Gordeyev and Thomas, 2000].

Because the direct relationship between the POD modes and the coherent structures is somewhat contested, the POD is used to provide a set of basis functions with which a low-dimensional subspace is identified on which a dynamical model of coherent structures can be constructed by projection of the governing equations [Holmes et al., 1996; Nobach et al., 2007]. Nevertheless, POD is a common tool in fluid dynamics in order to identify energetically relevant events. It

can be thought of as a set of basis flows, which allows an optimal description of many fluid flows. Apart from providing an optimal representation of the fluid flow, POD is also used as reconstruction or outlier replacement tool for measured flow fields. Accordingly it is used in order to fill in missing information of erroneous, gappy data to recover the complete flow field [Everson and Sirovich, 1995; Raben et al., 2012].

2.2.2 Basic principles of proper orthogonal decomposition

In the following, the POD is introduced together with some of its properties. The introduction complies with the introductions given in Nobach et al. [2007] and Chatterjee [2000]. Further information can be gained from Holmes et al. [1996] and Berkooz et al. [1993].

A function $u(\boldsymbol{x}, t)$, which is possibly vector valued and defined over some domain of interest $\mathcal{D} = \Omega \times [0, T]$ with $\Omega \subset \mathbb{R}^3$, can be approximated as a finite sum in the variable-separated form

$$u(\boldsymbol{x}, t) \approx \sum_{k=1}^{K} \alpha_k(t)\phi_k(\boldsymbol{x}) \; . \tag{2.6}$$

In this approximation, the $\phi_k(\boldsymbol{x})$ are some basis functions and the $\alpha_k(t)$ are coefficients. In fluid mechanics $\boldsymbol{x}$ is normally seen as spatial variable ($\boldsymbol{x} = (x, y, z) \in \Omega$) and t as temporal variable. The expectation is that the approximation becomes better with increasing K and is exact in the limit $K \to +\infty$.

There exist many basis functions $\phi_k(\boldsymbol{x})$ that solve the approximation (2.6). One could choose for example familiar functions like the *Fourier series*, the *Legendre polynomials*, or the *Chebyshev polynomials*. An alternative approach could be to determine the functions $\phi_k(\boldsymbol{x})$ that are naturally intrinsic for the approximation of $u(\boldsymbol{x}, t)$. This particular approach corresponds to the POD.

The time functions $\alpha_k(t)$ depend on the choice of the basis functions $\phi_k(\boldsymbol{x})$ and, therefore, are different for different sets of basis functions. In the case of orthonormal basis functions, which implies that

$$\int_\Omega \phi_{k_1}(\boldsymbol{x})\phi_{k_2}(\boldsymbol{x})\mathrm{d}\boldsymbol{x} = \begin{cases} 1, & \text{if } k_1 = k_2 \\ 0, & \text{otherwise} \end{cases} \; , \tag{2.7}$$

the coefficients are given by

$$\alpha_k(t) = \int_\Omega u(\boldsymbol{x}, t)\phi_k(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \; . \tag{2.8}$$

Orthonormality would be a good condition for the chosen basis functions because then, $\alpha_k(t)$ only depends on $\phi_k(\boldsymbol{x})$ and not on the other ϕ. A second condition is, that the approximation (2.6) is as good as possible in a least square sense for any chosen K. This means that the first two basis functions yield the best possible two term approximation and the first three basis functions yield the best possible three term approximation and so on.

Consider a data set, which consists of M realizations of $u(\boldsymbol{x}, t)$ at N different instants of time,

such as for instance a set of velocity vector fields. Then the optimal basis of this data is found by solving

$$\min \sum_{i=1}^{N} \|u(\boldsymbol{x}, t_i) - \sum_{k=1}^{K} \alpha_k(t_i)\phi_k(\boldsymbol{x})\|_2^2 \tag{2.9}$$

with the L^2-norm $\|\cdot\|_2$. The dataset can be arranged in a matrix $A \in \mathbb{R}^{M\times N}$ by transforming the data obtained at each instant of time into a column vector. The matrix is given by

$$A = \begin{pmatrix} u(\boldsymbol{x}_1, t_1) & u(\boldsymbol{x}_1, t_2) & \cdots & u(\boldsymbol{x}_1, t_N) \\ u(\boldsymbol{x}_2, t_1) & u(\boldsymbol{x}_2, t_2) & \cdots & u(\boldsymbol{x}_2, t_N) \\ \vdots & \vdots & \vdots & \vdots \\ u(\boldsymbol{x}_M, t_1) & u(\boldsymbol{x}_M, t_2) & & u(\boldsymbol{x}_M, t_N) \end{pmatrix}. \tag{2.10}$$

According to Nobach et al. [2007], a practical method in order to solve the minimization problem (2.9) is to use the *singular value decomposition* (SVD) described in the next section.

2.2.3 The singular value decomposition

The *singular value decomposition* (SVD) is a factorization of any real or complex matrix. Let A be a real $M \times N$ matrix ($A \in \mathbb{R}^{M\times N}$). Then the SVD is given by [Nobach et al., 2007]

$$A = U\Sigma V^T \tag{2.11}$$

where U is an orthogonal $M \times M$ matrix and V an orthogonal $N \times N$ matrix (i.e., $UU^T = I_M$ and $VV^T = I_N$, with I_N and I_M being the $N \times N$ and $M \times M$ identity matrix, respectively). A proof of the SVD can be found in Golub and Van Loan [1996]. The diagonal Matrix Σ is of size $M \times N$ and contains the *singular values* σ on its diagonal. These singular values are sorted in decreasing order $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r \geq 0$ with $r = \min(M, N)$. The rank of A equals the number r of its non-zero singular values. The i-th column of V is called the i-th right singular vector $\boldsymbol{v}_i$ and the i-th column of U is called the i-th left singular vector $\boldsymbol{u}_i$ with $i \in [1, \ldots, r]$.

It is shown in Nobach et al. [2007] that the left singular vectors of A correspond to the eigenvectors of AA^T and the right singular vectors of A correspond to the eigenvectors of A^TA. Furthermore, the singular values are equal to the square root of the eigenvalues, i.e., $\sigma_i = \sqrt{\lambda_i}$. This can easily be seen, if one computes the eigenvalue decomposition of AA^T that is $AA^T = W\Lambda W^{-1} = W\Lambda W^T$ with the orthogonal $M \times M$ matrix W and compares it to $AA^T = U\Sigma V^T V\Sigma U^T = U\Sigma^2 U^T$. This shows that the squared singular value matrix corresponds to the eigenvalue matrix and the eigenvector matrix corresponds to the matrix containing the left singular vectors, i.e., $\Sigma^2 = \Lambda$ and $W = U$, respectively. For A^TA one can proceed analogously. Further information about the relation between POD and SVD can be found in Fahl [2000] and in Volkwein [1999].

For a geometric illustration of the SVD, the $M \times N$ data matrix A can be seen as a list of

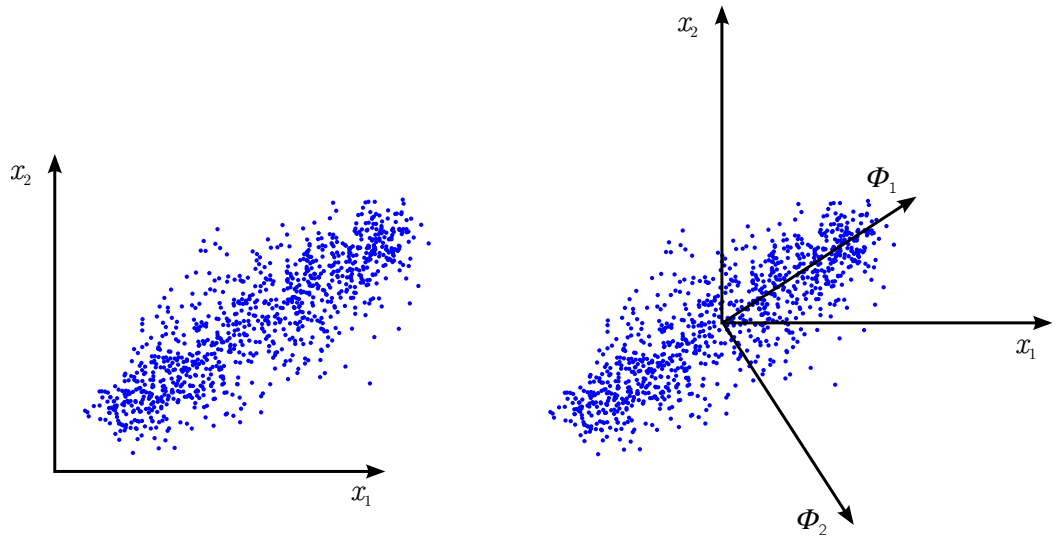

Figure 2.1: Geometric illustration of the SVD. *Left*: The data point cloud. *Right*: Mean shifted point cloud and new optimal basis.

coordinates of N points $P_1, P_2, \dots P_N$ with $P_i = (p_{i_x}, p_{i_y})^T$. The data points are illustrated in Figure 2.1 on the left as point cloud in the two dimensional coordinate system. Applying the POD or rather the SVD on the data matrix A yields a new optimal basis system with the two orthonormal basis vectors ϕ_1 and ϕ_2. Prior to the application of the SVD the data matrix is mean shifted by subtracting from each row of A the mean of that row. Therefore, the data is centered around the origin of the coordinate system. Optimality of the basis system implies that the first basis vector points in direction of the largest variance and the second basis vector is perpendicular to the first one and points in direction of the second largest variance. The mathematical concept of the SVD can geometrically be interpreted as a rotation of the original basis into a new coordinate system whose orthogonal axes coincide with the axes of inertia of the data [Nobach et al., 2007]. The mean shifted data is displayed together with the new rotated basis in Figure 2.1 on the right.

2.2.4 Proper orthogonal decomposition applied on a flow field

The following example shows, how POD modes of a fluid flow field typically look like. It further shows, how these POD modes can be used to approximate the flow field with a linear combination of the modes and some coefficients. Therefore, a POD is performed on the flow field data and a flow field patch, which is a local part of the flow field, is reconstructed with various numbers of POD modes. The flow field used for this example is the simulated velocity field of a 2D turbulent sequence of size $256 \times 256\,\mathrm{px}^2$, which is described in detail in Section 5.2.2. A sample set of 2000 flow field patches of size 31×31 was randomly chosen from the flow field. Performing a POD by means of SVD on the data set yielded 1922 ($= 2 \cdot 31 \cdot 31$) POD modes of size 31×31.

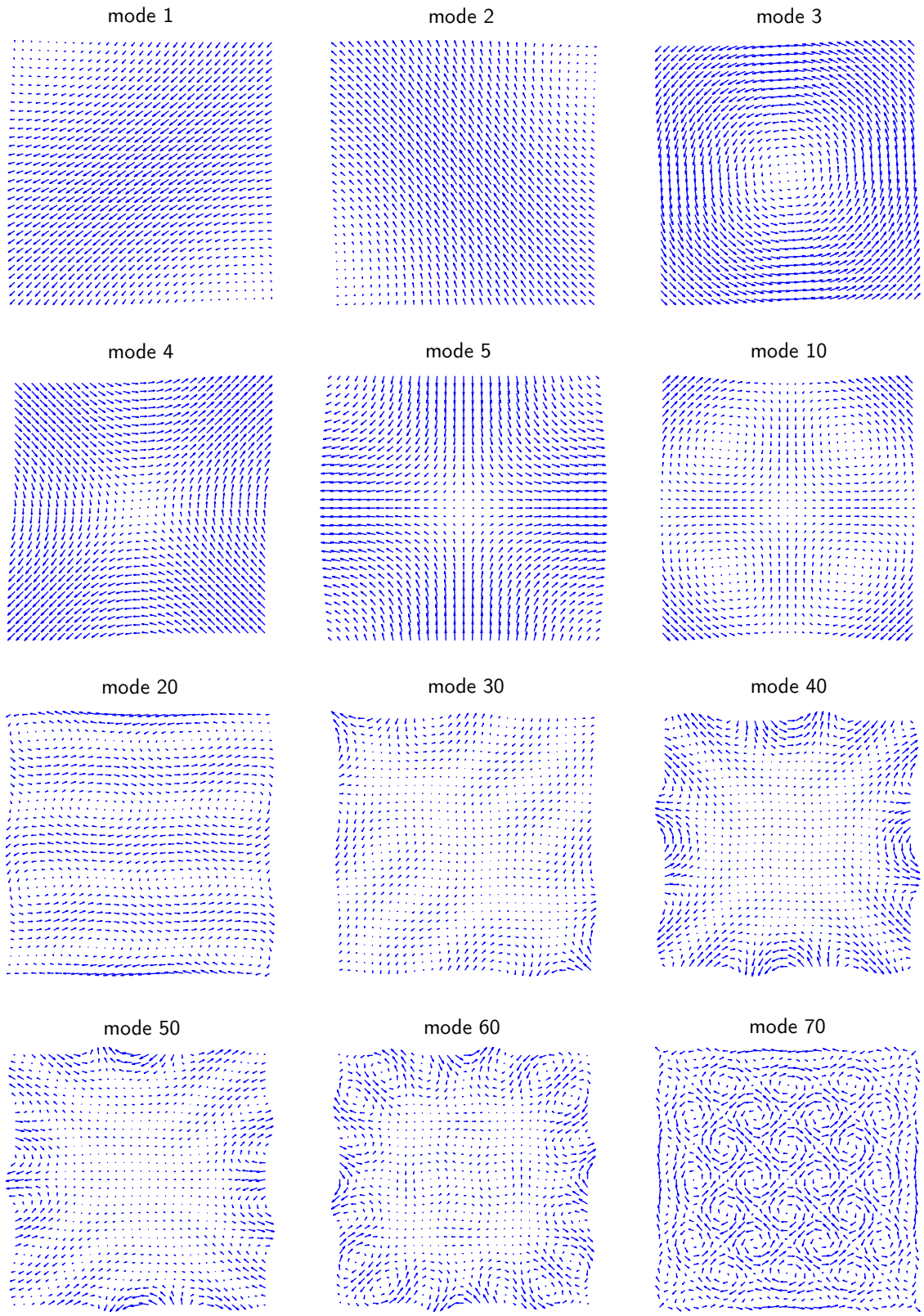

Figure 2.2: Selection of POD modes. Shown are the modes of size 31×31 obtained from the 2D turbulent sequence described in Section 5.2.2.

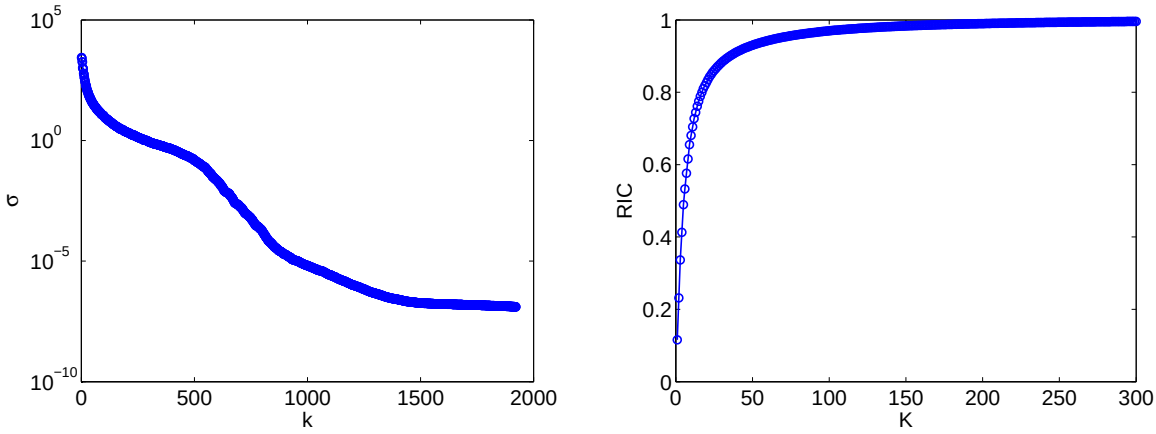

Figure 2.3: *Left*: Plot of the singular values σ in dependence of the components k. *Right*: Plot of the relative information content RIC in dependence of K.

These POD modes are also called *motion models*, or *eigenflows*. A detailed description on how motion models are derived from flow fields is given in Section 4.1.2. Figure 2.2 shows several of the determined POD modes. The first modes represent large scale structures and carry more energy or information than higher modes. For higher POD modes, the scales of the contained flow structures are getting smaller and for very high modes the structures belong more or less to noise. This fact is shown in the left plot of Figure 2.3, where the obtained singular values are depicted on a semi logarithmic scale. Each singular value represents the energy or information content of the belonging POD mode. The values decrease very fast for growing components k.

Following Nobach et al. [2007], the total number K of basis functions, which is needed to represent a fraction $\delta \in [0, 1]$ of the information contained in the original flow field patch, is given by the *relative information content* (RIC) defined by

$$\mathrm{RIC}(K) := \frac{\sum_{k=1}^{K} \sigma_k}{\sum_{k=1}^{2N} \sigma_k} \,. \tag{2.12}$$

This is simply the normalized part of the whole information contained in the first K basis functions. On the right side of Figure 2.3, the RIC is shown as a function of the number of used components K. To be clearly represented, only the first part of the RIC up to $K = 300$ is shown in the plot since the function approaches one quite fast. The plot indicates that 99 % of the whole information is contained within the first 10% of all basis functions. This means, that a flow field patch, which contains 99% of the total information of the original patch, can be approximated by using the first 200 modes. Since POD modes corresponding to small singular values carry only little information, they can be neglected. In order to determine the smallest number of basis functions K for which the RIC is larger or equal to a particular δ, the following

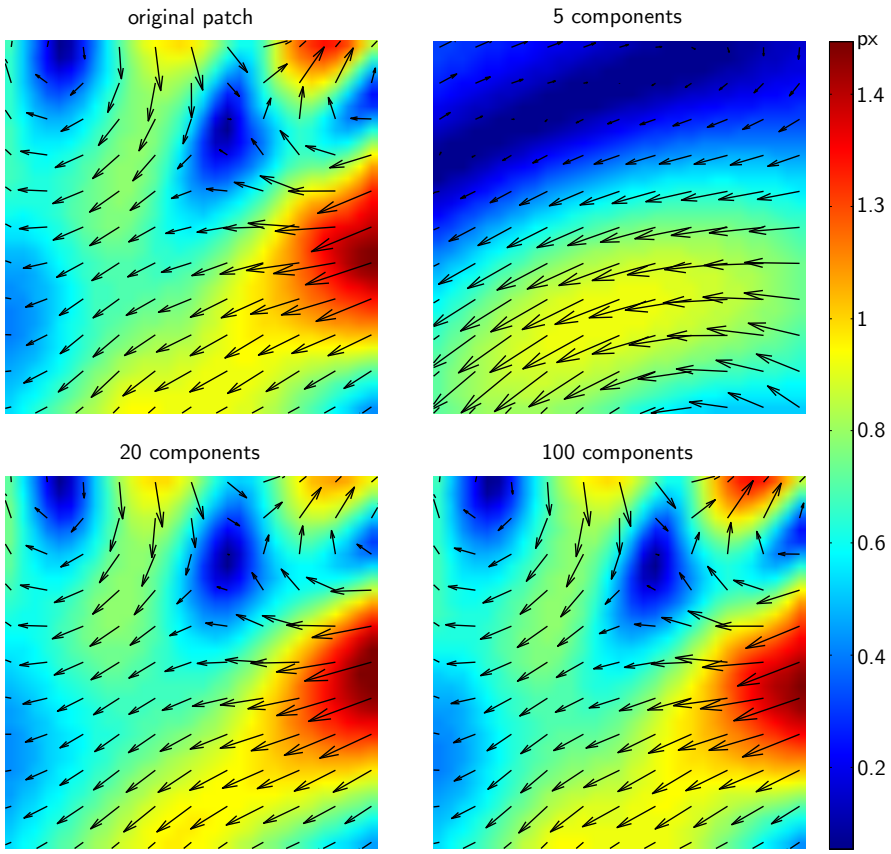

Figure 2.4: Original patch as well as the reconstructions taken from the 2D turbulent sequence. Top left: Original patch. Top right: Reconstruction with 5 modes. Bottom left: Reconstruction with 20 modes. Bottom right: Reconstruction with 100 modes. The magnitude of the velocity is color coded.

equation must be solved

$$K = \arg\min(\mathrm{RIC}(K); \mathrm{RIC}(K) \geq \delta) \; . \tag{2.13}$$

For the reconstruction of the flow field patch, the linear combination of coefficients α_k and POD modes ϕ_k of Equation (2.6) is used. Therefore, each coefficient α_k is computed by the scalar product of the original patch and the k-th POD mode ϕ_k, as shown in Equation (2.8). In Figure 2.4, one particular flow field patch of the 2D turbulent sequence is shown together with three reconstructions of this patch. For each reconstruction, a different number of POD modes was used. According to Figure 2.3, a reconstruction with the first 5 modes contains 50 % of the total information. In this case, the reconstructed patch looks in large areas very different than the original patch. Especially the upper half shows strong disparities. The reconstructed patch with the first 20 components contains approximately 85 % of the information and the similarity to the original patch is already quite good. However, there are small but visible differences. By using the first 100 modes, the RIC rises to 97 % and there are no differences observable by eye between the reconstructed and the original patch.

2.3 Mathematical preliminaries

The following mathematical section gives a short recapitulation of some methods and tools which are used in this thesis. The intention is to help the reader to understand the concepts presented in the following chapters. The description covers the main aspects and further information can be found in the literature quoted. In particular, Section 2.3.1 addresses how to minimize functionals by solving the corresponding Euler-Lagrange equation. Therefore, the solution of large systems of linear equations is sometimes required as described in Section 2.3.2. Section 2.3.3 introduces the solution of overdetermined systems of linear equations. Finally, the concept of linear filtering, which is needed for many image processing tasks, is introduced in Section 2.3.4.

2.3.1 Calculus of variation

The theory of *calculus of variations* deals with the optimization, i.e., minimization and maximization of *functionals*. A functional can be understood as a function of a function. More precisely, a functional is defined as a function from a vector space, which is often a space of functions, into its underlying scalar field.

Consider the functional of integral type over some domain $\Omega \in \mathbb{R}^2$

$$J(u) = \int_\Omega \mathcal{L}\left(x, y, u(x,y), u_x(x,y), u_y(x,y)\right) \mathrm{d}\boldsymbol{x} \tag{2.14}$$

on the set of twice continuously differentiable functions $u = u(x,y)$ with known boundary values on $\partial\Omega$ and $\boldsymbol{x} = (x,y)^T \in \Omega$. Here and in the following, u_x and u_y denote the derivatives of u with respect to x and y, respectively. A task of calculus of variations is now to find a function u which minimizes the functional $J(u)$. A minimizing function is found by solving the *Euler-Lagrange* equation given by

$$\frac{\partial \mathcal{L}}{\partial u} - \frac{\mathrm{d}}{\mathrm{d}x}\frac{\partial \mathcal{L}}{\partial u_x} - \frac{\mathrm{d}}{\mathrm{d}y}\frac{\partial \mathcal{L}}{\partial u_y} = 0\,. \tag{2.15}$$

Further information on calculus of variations including a proof of Equation (2.15) is given for instance in Lebedev and Cloud [2003]. Solving the Euler-Lagrange equation often comes down to solving a large system of linear equations. Therefore, in Section 2.3.2 some solution concepts for large systems of linear equations are introduced.

2.3.2 Fixed point iteration

As explained before, the optimization process often leads to a large system of linear equations such as:

$$A\boldsymbol{x} = \boldsymbol{b} \tag{2.16}$$

with the real valued coefficient matrix $A = (a_{ij}) \in \mathbb{R}^{n\times n}$, the right hand side vector $\boldsymbol{b} = (b_i) \in \mathbb{R}^n$, and the solution vector $\boldsymbol{x} = (x_i) \in \mathbb{R}^n$. This system of equations is called quadratic because

the number of unknowns embodied by the components x_i equals the number of equations. If the equations are linearly independent, i.e., $\text{rank}(A) = n$, Equation (2.16) has a unique solution. Otherwise, if $\text{rank}(A) < n$, infinitely many solutions exist. If the number of equations exceeds the number of unknowns, which is the case for $A \in \mathbb{R}^{m \times n}$ and $m > n$, no exact solution exists, but usually an approximated solution can be found as shown in Section 2.3.3.

In order to solve the system of linear equations (2.16), many different methods can be used. Basically there are two different groups of techniques. On the one hand, there are *direct* methods such as the *Gaussian elimination* method where the solution is determined within a finite number of calculation steps up to some rounding error. On the other hand, there are *iterative* methods which successively determine a sequence of vectors $\left(\boldsymbol{x}^{(q)}\right)_{q \in \mathbb{N}}$, which tend to better and better approximations of the exact solution $\boldsymbol{x}$ for increasing q.

For large systems of linear equations with $n \gg 1000$, direct methods become very impractical and slow due to memory requirements and computational costs. The number of required operations is of the order of $\frac{2}{3}n^3$ [Quarteroni et al., 2000]. Especially in the case of large sparse matrices A with a number of non-zero entries on the order of n, iterative methods are the better choice to solve the problem. For these methods, the number of operations per iteration step is proportional to the number of non-zero elements and is, therefore, on the order of n for sparse systems.

The focus is on iterative methods because the systems of linear equations that occur within this work are in fact large and sparse. They are also called fixed point methods. For a function f, a fixed point iteration defined by

$$x^{(q+1)} = f\left(x^{(q)}\right) \tag{2.17}$$

is often used to approximate the fixed point x of the function given by

$$f(x) = x\ . \tag{2.18}$$

Of course the fixed point iteration is only successful if the sequence $x^{(1)}, x^{(2)}, \ldots$ converges to the fixed point x, which is the case, if the spectral radius given by the maximum *eigenvalue* of the iteration matrix is smaller than one. An extensive description of convergence criteria of iterative methods is given for example in Quarteroni et al. [2000].

In the following, some iterative methods are introduced. For a more comprehensive review of the topic see for instance Quarteroni et al. [2000] or Schwarz and Köckler [2011].

Jacobi method

The simplest iterative solution of the system of linear equations (2.16) is given by the *Jacobi* method. Accordingly, the matrix A is decomposed into a diagonal matrix D and a reminder R. It is: $A = D + R$. This leads to the reformulated problem $D\boldsymbol{x} = \boldsymbol{b} - R\boldsymbol{x}$ which is solved

iteratively by

$$\boldsymbol{x}^{(q+1)} = D^{-1}\left(\boldsymbol{b} - R\boldsymbol{x}^{(q)}\right) \tag{2.19}$$

with the iteration parameter q. With every iteration the solution $\boldsymbol{x}^{(q+1)}$ is updated starting with an initial value $\boldsymbol{x}^{(1)}$. In a point-based formulation the solution is given by

$$x_i^{(q+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{\substack{j=1 \\ j\neq i}}^{N} a_{ij}x_j^{(q)}\right) \quad , \quad i = 1, \ldots, n \; . \tag{2.20}$$

In simple words, the Jacobi method utilizes constantly updated values $x_j^{(q)}$ with $j \neq i$ to determine $x_i^{(q+1)}$ until convergence is reached.

Gauss-Seidel method

At the time of the determination of $x_i^{(q+1)}$, the values $x_j^{(q+1)}$ with $j < i$ are already known and could be used for the calculation. This is the principle of the *Gauss-Seidel* method. With a decomposition of A into a diagonal matrix D, a strictly lower triangular matrix L, and a strictly upper triangular matrix U given by $A = D + L + U$, the problem (2.16) can be reformulated as $(D + L)\,\boldsymbol{x} = (\boldsymbol{b} - U\boldsymbol{x})$. The iterative solution is then given by

$$\boldsymbol{x}^{(q+1)} = (D + L)^{-1}\left(\boldsymbol{b} - U\boldsymbol{x}^{(q)}\right) \; . \tag{2.21}$$

The strictly lower triangular matrix L addresses the values $x_j^{(q+1)}$ with $j < i$ from the current iteration step that are already known. The strictly upper triangular matrix U addresses the values $x_j^{(q)}$ with $j > i$ known from the previous iteration step. In point-based notation the solution is given by

$$x_i^{(q+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(q+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(q)}\right) \quad , \quad i = 1, \ldots, n \; . \tag{2.22}$$

Compared to the Jacobi method the Gauss-Seidel method converges faster and, therefore, fewer iterations steps are required to reach a fixed point. In the case that A is symmetric and positive definite (i.e., all eigenvalues are positive), convergence is guaranteed.

Successive over-relaxation

The *successive over-relaxation* (SOR) method leads to an additional speed up of the convergence. In fact it is a weighted linear combination of the value $x_i^{(q)}$ of the previous iteration and the Gauss-Seidel value $\tilde{x}_i^{(q+1)}$ of the actual iteration step:

$$x_i^{(q+1)} = \omega\tilde{x}_i^{(q+1)} + (1 - \omega)x_i^{(q)} \tag{2.23}$$

with the relaxation parameter $\omega \in (0, 2)$. For $\omega = 1$, this comes down to the simple Gauss-Seidel method. If $\omega < 1$, which is in fact called under-relaxation, a solution that would normally slightly diverge may be stabilized. Choosing $\omega > 1$ may significantly accelerate the convergence compared to the Gauss-Seidel method and is called over-relaxation. The choice of an optimal relaxation parameter ω_{opt} is not quite easy and depends strongly on the specific problem [Quarteroni et al., 2000]. It is therefore often chosen empirically.

Using once again the decomposition $A = D + L + U$, the matrix notation of the SOR method is given by

$$\boldsymbol{x}^{(q+1)} = (D + \omega L)^{-1} \left(\omega \boldsymbol{b} + ((1 - \omega) D - \omega U) \boldsymbol{x}^{(q)} \right) . \tag{2.24}$$

In point-based notation the iteration instruction can be formulated as

$$x_i^{(q+1)} = (1 - \omega) x_i^{(q)} + \frac{\omega}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(q+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(q)} \right) , \quad i = 1, \ldots, n . \tag{2.25}$$

Outer fixed point iteration

In the case the system of equations consists of non-linear equations, it can be decomposed into a series of linear problems, which then can be solved by the standard linear techniques presented in the previous sections. This method is called *lagged diffusivity* [Bruhn, 2006; Chan and Mulet, 1999]. The system of non-linear equations can be described by

$$A(\boldsymbol{x}) = \boldsymbol{b} \tag{2.26}$$

with the non-linear operator $A(\boldsymbol{x})$, which can be decomposed into

$$A(\boldsymbol{x}) = B(\boldsymbol{x})\boldsymbol{x} + \boldsymbol{c}(\boldsymbol{x}) . \tag{2.27}$$

Thereby, matrix $B(\boldsymbol{x}) \in \mathbb{R}^{n \times n}$ and vector $\boldsymbol{c}(\boldsymbol{x}) \in \mathbb{R}^n$ are again non-linear operators. Assuming that the matrix $B(\boldsymbol{x})$ is symmetric and positive definite the fixed point iteration

$$\boldsymbol{x}^{(q+1)} = \left(B\left(\boldsymbol{x}^{(q)}\right) \right)^{-1} \left(\boldsymbol{b} - \boldsymbol{c}\left(\boldsymbol{x}^{(q)}\right) \right) \tag{2.28}$$

is solved while keeping a fixed argument $\boldsymbol{x}^{(q)}$ for the non-linear operators. Thus, the non-linear expressions are evaluated at the old time step q. At each time step a remaining linear system of equations has to be solved, which can be done iteratively by one of the introduced methods.

2.3.3 Method of least squares

In the case that the system of linear equations (2.16) is overdetermined, the method of *least squares* can be used to derive a solution. This short introduction of the method follows the monograph of Van Huffel and Vandewalle [1991]. The underlying problem is, to find a solution

vector $\boldsymbol{x} \in \mathbb{R}^n$ of the system of linear equations

$$A\boldsymbol{x} = \boldsymbol{b} \tag{2.29}$$

with given data matrix $A \in \mathbb{R}^{m \times n}$ and vector of observations $\boldsymbol{b} \in \mathbb{R}^m$. In the case that there are more equations than unknowns, i.e., $m > n$, the system of equations is overdetermined and has typically no exact solution. Therefore, it can be written as $A\boldsymbol{x} \approx \boldsymbol{b}$. There exist several approximated solutions of this problem but the question is, which of these solutions is the best? The least squares solution of $A\boldsymbol{x} \approx \boldsymbol{b}$ is derived by solving

$$\min \|A\boldsymbol{x} - \boldsymbol{b}\|_2 \tag{2.30}$$

with the L^2-norm $\|\cdot\|_2$. The minimizing vector is denoted by $\boldsymbol{x}^{\text{ls}}$ and is given by the so-called *normal equation*:

$$\boldsymbol{x}^{\text{ls}} = \left(A^T A\right)^{-1} A^T b \,. \tag{2.31}$$

Of course the solution is only defined if the inverse matrix $\left(A^T A\right)^{-1}$ exists. In the case of a singular matrix, the *Moore-Penrose pseudoinverse* [Ben-Israel and Greville, 2003] can be used in order to find the least squares solution. The Equation (2.31) is obtained by considering the principles of *calculus* which state that the minimum of a function can be found by setting its derivatives to zero and solve the corresponding equations.

It is assumed that only the observation vector $\boldsymbol{b}$ may be erroneous, whereas, the data points given by A are error free. This means that the observation points $\boldsymbol{b}$ may differ from the correct points $\boldsymbol{b}_0$ by $\Delta\boldsymbol{b}$.

The idea of the method of least squares is to minimize the sum of the squared residual $\boldsymbol{r}$, which can be interpreted as the deviation of the vector $\boldsymbol{b}^{\text{ls}}$ obtained from the least squares solution $\boldsymbol{x}^{\text{ls}}$, from the real observation vector $\boldsymbol{b}$:

$$\boldsymbol{r} = \boldsymbol{b} - A\boldsymbol{x}^{\text{ls}} = \boldsymbol{b} - \boldsymbol{b}^{\text{ls}} \,. \tag{2.32}$$

A graphical interpretation of the method of least squares for a one parameter estimation $\alpha x = \beta$ is shown in the left plot of Figure 2.5. The plot contains several data points (a_i, b_i) with $i \in [1, \ldots, m]$. The solution x is given by the slope of the line fitted to the data points. The least squares solution x^{ls} is the one that minimizes the sum of the quadratic differences of the real observations $\boldsymbol{b}$ and the predicted observations $\boldsymbol{b}^{\text{ls}}$, i.e., the quadratic vertical distances of data points and the line.

Given that the dependent variables $\boldsymbol{b}$ as well as the independent variables given by the coefficients of A are erroneous, the method of *total least squares* might yield better results. This method considers independent and identically distributed errors described within the deviations $\Delta\boldsymbol{b}$ and ΔA of both types of variables. The method of total least squares can be formulated as

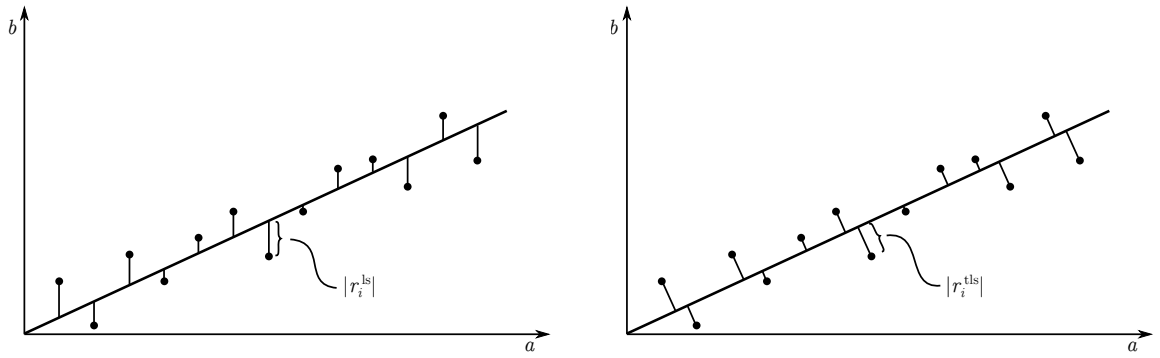

Figure 2.5: Geometric interpretation of the method of least squares (*left*) and the method of total least squares (*right*). Within the least squares solution the vertical distances and within the total least squares solution the orthogonal distances are minimized.

the minimization problem

$$\min \left\| [A, \boldsymbol{b}] - \left[A^{\text{tls}}, \boldsymbol{b}^{\text{tls}} \right] \right\|_F \tag{2.33}$$

using the augmented matrices $[A, \boldsymbol{b}]$ and $\left[A^{\text{tls}}, \boldsymbol{b}^{\text{tls}}\right]$ and the *Frobenius norm* $\|\cdot\|_F$, which is defined by $\|A\|_F = \sqrt{\sum_i \sum_j a_{ij}}$. The total least squares solution $\boldsymbol{x}^{\text{tls}}$ exactly solves the system of linear equations:

$$A^{\text{tls}} \boldsymbol{x}^{\text{tls}} = \boldsymbol{b}^{\text{tls}} \tag{2.34}$$

with the corrected total least squares variables $\left[A^{\text{tls}}, \boldsymbol{b}^{\text{tls}}\right] = [A, \boldsymbol{b}] + [\Delta A, \Delta \boldsymbol{b}]$.

According to Van Huffel and Vandewalle [1991], the total least squares solution of (2.29) is given by

$$\boldsymbol{x}^{\text{tls}} = \left(A^T A - \sigma_{\min} I \right)^{-1} A^T b \,. \tag{2.35}$$

Here, $\sigma_{\min}$ denotes the smallest *singular value* (cf. Section 2.2.3) of the augmented matrix $[A, b]$ and I denotes the identity matrix. According to the method of least squares a graphical interpretation of the method of total least squares is given in the right plot of Figure 2.5.

2.3.4 Linear filtering

Filter operations are very important for image processing. They are used for instance to sharpen or blur image details, to remove noise, or to determine gradient images. Therefore, the images are convolved with a *filter kernel* which is also called *impulse respond function* [Jähne, 2005]. It is a neighborhood operation which connects the pixel values in the vicinity of a given pixel with the pixel values of the filter kernel to determine an output value. Type and size of the kernel account for the character of the filter operation. A common filter kernel used for smoothing operations and noise reduction is for instance the Gaussian kernel

$$K_\sigma(x, y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{2.36}$$

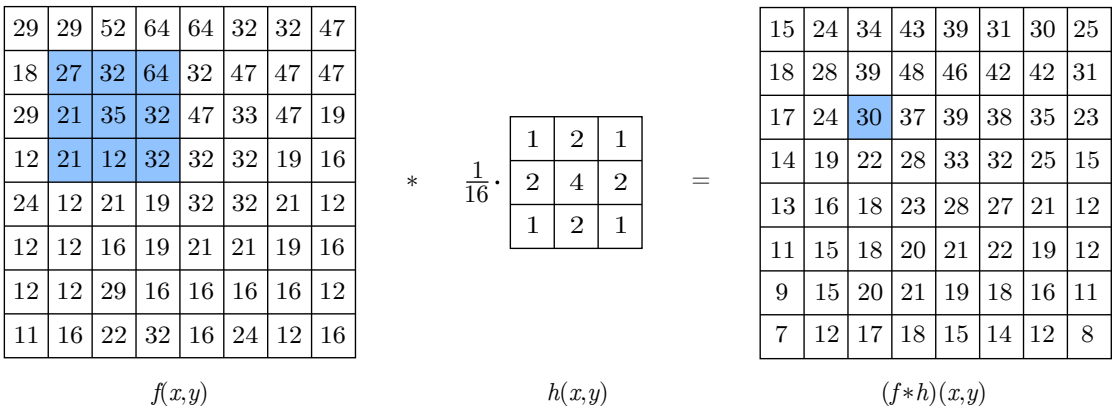

Figure 2.6: Illustration of the filtering operation (convolution). The discrete function $f(x, y)$ is convolved with the discrete filter kernel $h(x, y)$. The blue pixel locations in f yield together with the filter kernel the blue pixel location in $f * h$.

with standard deviation σ. Due to the discrete nature of the images, the discrete version of the convolution is used. For the discrete functions f and h it is given by [Szeliski, 2011]

$$(f * h)(i, j) = \sum_{(k,l)\in\mathcal{N}} f(i - k, j - l)h(k, l) \tag{2.37}$$

where (i, j) denotes a discrete pixel location in the image domain and (k, l) a discrete pixel location in the filter kernel domain $\mathcal{N}$. A graphical illustration of the convolution is shown in Figure 2.6. It can be performed by first mirroring the filter kernel on its center of symmetry and then sliding it pixelwise over the image. At each pixel location the linear combination of filter coefficients and image pixels yields the current value.

One problem of the filtering operation is the calculation of the boundary pixels because here, the filter kernel extends beyond the image boundaries. To overcome this problem, the image is normally augmented either by zeros (zero padding) or another fixed value, by repeating the values of the nearest edge pixel (replicate), by reflecting pixels at the image edge (mirror), or by assuming a periodic input array (circular). This ensures, that the filtered image has the same size as the initial image. Nevertheless, the boundary pixel values might not be very accurate.

An important property of many filter kernels used in image processing regarding the computation time is their *separability*. If a 2D filter kernel is separable, it can be separated into two 1D filter kernels, which are applied consecutively. Without separation, the convolution requires about N^2 operations per pixel, with N being the size (width or height) of the filter kernel. Splitting the convolution in two successive convolutions requires $2N$ operations [Szeliski, 2011]. Within this project, mainly the filter kernels optimized for optical flow computations introduced by Scharr [2007] were used.

3 Fluid motion detection

In this chapter, the basic principles of fluid motion detection are presented. The experimental setup of image-based fluid flow measurements is explained in Section 3.1. In Section 3.2, the term *optical flow* is introduced and its relationship to the *motion field* and the *displacement field* is described. In order to determine the fluid flow field from image sequences, many different image processing techniques can be applied. Most commonly used in the research of fluid dynamics is a correlation-based approach described in Section 3.3. Since the methods developed in this thesis belong to the gradient-based optical flow techniques, which originate from computer vision, the basic principles of these approaches are presented in Section 3.4. In order to determine the quality of the estimated displacement fields, so-called error measures introduced in Section 3.5 are needed.

3.1 The measurement of fluid flows

Fluid flow measurements can be divided into *qualitative* and *quantitative* techniques. A qualitative technique is for instance the simple flow visualization which has a long tradition in fluid mechanics. At the beginning of the 20th century, Ludwig Prandtl studied unsteady, separated flows behind objects by utilizing flow visualization techniques in a water channel [Raffel et al., 2007]. His observations [Prandtl, 1904] led to the introduction of the concept of the boundary layer [Prandtl and Betz, 2010]. The aim of flow visualization is to make the movement of transparent liquids and gases visible to the observer. In order to do so, some tracer material such as smoke, dye, or small artificial particles is added to the fluid, acting as an indicator of fluid motion. It is crucial that the difference between the motion of the added tracer material and the fluid is negligible, since only the motion of the tracer is observable but the motion of the fluid is of interest. Different flow visualization techniques are presented in Merzkirch [2007].

The quantitative flow measurements can be classified into point-wise measurements including measurements with a pitot tube, a thermal anemometer, or a laser Doppler velocimeter, and in whole field techniques, which extend the qualitative flow visualization methods. Advantages of the highly developed point-wise techniques are their accuracy and rapid frequency response [Adrian and Westerweel, 2011]. However, the limitation is that information can be gained only at one specific location at one instant of time and, therefore, no information about the surrounding flow structure can be obtained. Single-point measurements do not allow to take instantaneously spatial derivatives of the flow which would be extremely important for many interesting fluid

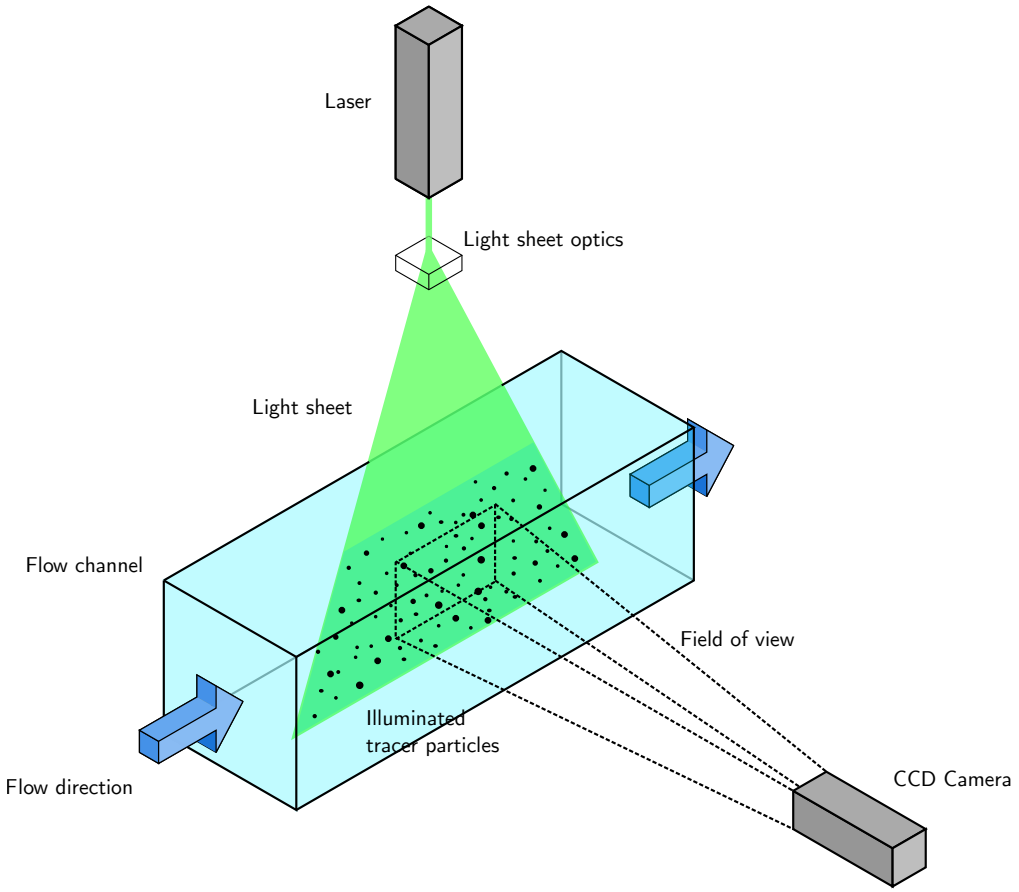

Figure 3.1: Sketch of the experimental setup of standard PIV. The fluid, seeded with tracer particles, is illuminated by a pulsed laser in a plane by applying light sheet optics. The light, which is reflected by the particles, is then observed by a camera.

mechanical properties such as for instance the vorticity [Adrian and Westerweel, 2011].

Whole field techniques are able to measure the complete velocity vector field simultaneously. Typically, they are optical methods originating from flow visualization and are, therefore, non-intrusive which means that the flow is not influenced by the measurement as it is for instance the case for measurements with the pitot tube. Here, the flow may be disturbed by the instrument. A very common whole field technique is *particle image velocimetry* (PIV), where an image sequence of the fluid and additional tracer material is recorded. The fluid velocity is then estimated from the sequence applying image processing techniques. The experimental setup is sketched in Figure 3.1. In most applications the fluid is seeded with appropriate tracer particles, e.g., silver coated hollow glass spheres. The fluid is illuminated in a plane by a thin laser light sheet and the light reflected by the tracer particles is captured on a CCD-sensor of a camera. In order to determine the motion, the fluid is illuminated and recorded at least twice in a short period of time. From the particle shift between subsequent images, a displacement field can be estimated as explained in Section 3.3 and 3.4.

Apart from PIV also other whole field measurement techniques tailored towards specific applications are utilized. One example is *particle tracking velocimetry* (PTV) which is used to track individual particles belonging to single fluid parcels. This is useful, if a Lagrangian description of the flow is of interest. Another whole field technique is *laser induced fluorescence* (LIF) where a particular transition of some LIF-active molecules is excited by a laser light sheet and the emitted light is observed by a camera. LIF is often used to study combustion processes, sometimes even in combination with PIV.

3.2 The motion field

The complete velocity field of a fluid at one point in time is given by the 3D velocity vectors at every location of a volume. Since we are merely taking images of the scene, as explained in Section 3.1, we normally can only extract a 2D slice of the complete velocity field. Therefore, the 2D *motion field* is a projection of the 3D real object motion relative to the image sensor onto the image plane. Unfortunately, all information of motion perpendicular to the image plane is lost due to this projection. Yet, the motion field of fluids is of major interest to many researchers.

The beholder can observe only the so called *optical flow*, which is the apparent motion in the image plane given by the flow of the gray values in an image sequence. The optical flow has the dimension of a velocity and in the following it is denoted by the vector $\boldsymbol{u} = (u, v)^T$. The optical flow vector between two subsequent images is called *displacement vector* $\boldsymbol{d} = (d_1, d_2)^T$. It has the dimension of a length. A dense representation of displacement vectors, i.e., one vector per image pixel, is called *displacement field*. Dividing the displacement field by the time difference Δt between the two successive images yields a discrete approximation of the continuous *optical flow field*.

The optical flow field itself is an approximation of the 2D motion field. Equality between both fields is only given, if the irradiance of the imaged objects on the image plane stays constant while moving in the scene [Jähne, 2005]. However, in practice this is not always true and not every gray value change can be connected to real world motion and moreover not every object motion is depicted by a gray value change. A prominent example for the failure of optical flow is a rotating sphere and its motion field. A homogeneous textured, spinning sphere with constant illumination leads to zero optical flow, whereas a static sphere with varying illumination causes non-zero optical flow [Horn, 1986].

In the simplest case of determining the optical flow one assumes the constancy of brightness, which means that the brightness of a moving gray value distribution in an image sequence remains constant over time. This implies a constant homogeneous illumination of the imaged scene and no occlusion of the objects. Let $I_i(\boldsymbol{x})$ be the image brightness at location $\boldsymbol{x} = (x, y)^T$ of the i-th image of a sequence, which is proportional to the radiance received by a camera. The 2D image domain Ω constitutes of a sampling grid given by an orthogonal lattice and $\boldsymbol{x} \in \Omega$ is an arbitrary point in it. For ideal images (i.e., no camera noise) the assumption of the constancy

of brightness can be expressed as

$$I_i(\boldsymbol{x}) = I_{i+1}(\boldsymbol{x} + \boldsymbol{d}(\boldsymbol{x})) \; . \tag{3.1}$$

Considering whole images, Equation (3.1) states that the subsequent image I_{i+1} is a displaced version of the image I_i.

The determination of the displacement field $\boldsymbol{d}$ from Equation (3.1) is an inverse problem. Since this is a single equation with two unknowns, namely the two components of the displacement vector, the problem is ill-posed in the sense of Hadamard [1902]. However, by utilizing additional constraints it is possible to receive a solution. One possibility is to assume that the displacement field is locally constant in small image regions and to pool one solution from several equations. Correlation-based techniques such as PIV (cf. Section 3.3) provide only one vector per region as well as local optical flow methods (cf. Section 3.4.4). Another possibility is to use additional regularization terms such as for instance the smoothness of the solution as shown in Section 3.4.4.

3.3 Particle image velocimetry

In this section correlation-based motion estimation schemes for the determination of fluid motion are presented. These methods constitute the classical *particle image velocimetry* (PIV) approach. Since correlation-based techniques are most often used in fluid flow velocity measurements derived from PIV images they are often simply denoted as PIV methods. The technique is described comprehensively by Raffel et al. [2007] and by Adrian and Westerweel [2011].

3.3.1 The standard approach

The standard PIV approach is sketched in principles in Figure 3.2. In order to determine the displacement field, two subsequent gray value images of the PIV sequence are divided into interrogation windows and the cross correlation coefficient of two corresponding windows is computed. The cross correlation coefficient is a normalized version of the cross correlation function and is used to disable the influence of the particle brightness on the correlation peak height. The position of the peak of the correlation surface corresponds to the displacement vector associated with the center of the interrogation window. The cross correlation coefficient function for one particular window $\mathcal{W}_i$ is given by [Raffel et al., 2007]

$$C(\boldsymbol{d})\big|_{\mathcal{W}_i} = \frac{\sum\limits_{\boldsymbol{x}\in\mathcal{W}_i} I_1(\boldsymbol{x}) I_2(\boldsymbol{x}+\boldsymbol{d})}{\sqrt{\sum\limits_{\boldsymbol{x}\in\mathcal{W}_i} I_1^2(\boldsymbol{x}) \sum\limits_{\boldsymbol{x}\in\mathcal{W}_i} I_2^2(\boldsymbol{x})}} \tag{3.2}$$

where $\boldsymbol{d}$ represents the 2D displacement vector, $\boldsymbol{x}$ the two dimensional spatial vector, and I_1 and I_2 the first and the second mean subtracted frame of an image pair, respectively. By dividing

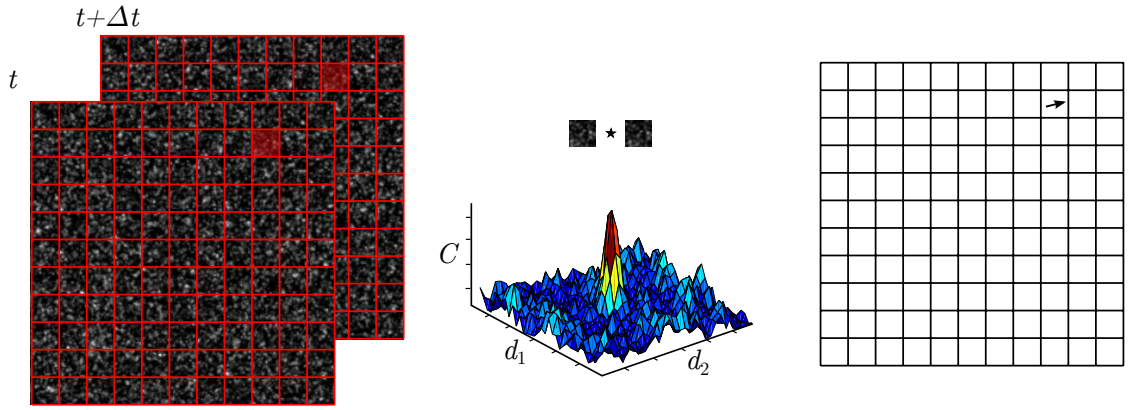

Figure 3.2: The principle of double frame single exposure PIV. Two subsequent images at time t and $t+\Delta t$ are divided into interrogation regions. The maximum of the cross correlation coefficient function of two corresponding interrogation regions (e.g., the red dyed squares) gives the displacement of the whole region and thus, one displacement vector in the region center.

the displacement vector $\boldsymbol{d}$ by the time difference Δt between two subsequent images one obtains the velocity vector.

To speed things up, the computational heavy cross correlation is usually replaced by a multiplication in Fourier space. This is feasible based on the *correlation theorem*, which states that the cross correlation of the two interrogation windows can be performed in Fourier space by a multiplication of the Fourier transform of one window with the complex conjugate Fourier transform of the other window.

3.3.2 Limits and extensions of the standard approach

Over the past decades a large variety of different methods have been proposed to improve the results and the computational performance compared to standard PIV [Adrian, 1991; Adrian and Westerweel, 2011; Raffel et al., 2007].

The standard PIV approach suffers from a number of shortcomings that limit the accuracy, the dynamic range, and the spatial resolution [Jähne et al., 2007]. These limitations are mainly a consequence of the finite size of the interrogation windows. A trade-off has to be found between large and small interrogation windows. Large windows can resolve large motions and provide good accuracy due to a high signal-to-noise ratio and are more robust to outliers. Small windows yield a better spatial resolution and are less affected by velocity gradients such as shear flows or vortices. To provide a robust evaluation of the cross-correlation the interrogation windows must contain a sufficient number of particles, which depends on the particle density, and thus, cannot be arbitrarily small. According to Keane and Adrian [1992], the number of corresponding particles in both interrogation windows has to be at least four to allow for an adequate detection reliability. However, usually the effective number of particles per window must be higher in

order to compensate in-plane and out-of-plane losses of particle pairs. Another constraint of the window size is the *one quarter rule*, which states that the size of the interrogation window has to be at least four times bigger than the maximum displacement [Keane and Adrian, 1990].

To circumvent these problems, a hierarchical recursive approach, which employs successively reduced correlation window sizes, can be used [Scarano and Riethmuller, 1999]. Since flows with strong velocity gradients lead to a sheared particle pattern in the second interrogation window resulting in a smaller and broader cross correlation coefficient peak, Scarano [2002] proposed to use a continuous window deformation technique.

Another limitation of PIV approaches is the *peak locking* effect, which means that displacement vectors are biased towards integer values. The effect can be attenuated by an increased particle size. The optimal particle diameter is slightly more than two pixels [Raffel et al., 2007].

The PIV methods described so far are two dimensional two component (2D2C) techniques, which means that only two of the three velocity components are determined in a plane of the measurement volume. However, for many turbulent flow applications the knowledge of the full 3D3C velocity field is required for a thorough investigation of the physical issues of interest. Hence, improved methods have been developed to determine the complete 3D3C velocity field with PIV methods, such as for instance holographic PIV [Hinsch, 2002] and tomographic PIV [Elsinga et al., 2006].

3.3.3 PIV algorithm used in this thesis

For the estimation of PIV velocity vector fields within the scope of this thesis, the software *fluere* (Version 1.3) created by Kyle Lynch was used. The program is available under the terms of the *GNU General Public License* (GPL). It is based on the iterative image deformation algorithm described by Scarano and Riethmuller [2000]. This algorithm enhances the matching performance of two interrogation windows by means of relative transformation between the windows. On the basis of an iterative prediction of the tracer motion, window offset and deformation are applied, accounting for the local deformation of the fluid continuum [Scarano and Riethmuller, 2000]. Apart from the standard correlation mode, fluere has also a multi-frame correlation mode, where a sliding ensemble correlation is used over a small kernel of images. The software also removes and replaces spurious vectors identified by a normalized median test. Overall, the algorithm allows for a wide dynamic range and reduces the effects of peak locking.

3.4 Optical Flow

Although, originally the approximation of the motion field itself was named optical flow (cf. Section 3.2) the estimation scheme to determine the optical flow field is often also simply called optical flow, especially global methods (cf. Section 3.4.4) are frequently referred to optical flow [Heitz et al., 2010]. The technique was developed roughly thirty years ago by the computer vision community. Path breaking work back then was done by Horn and Schunck [1981]. Until

today it remains to be one of the primary research topics in computer vision and the concepts are currently used for motion estimation as well as stereo vision.

Since the beginning of optical flow research there exist several different classes of motion estimation techniques [Beauchemin and Barron, 1995]. Most common are *differential* or *gradient-based* methods which build on spatio-temporal derivatives of the image intensity [Horn and Schunck, 1981; Lucas and Kanade, 1981]. Other techniques are for instance *region-based matching* [Anandan, 1989; Glazer et al., 1983], *frequency-based* methods [Adelson and Berger, 1985; Heeger, 1988], and *phase-based* methods [Fleet and Jepson, 1990; Waxmann et al., 1988].

Over the years the original optical flow methods have been significantly refined and improved. A variety of different methods is presented and compared in Barron et al. [1994] and more recently in Baker et al. [2011]. Further general information on motion estimation with optical flow is provided by Jähne et al. [2007], Haussecker and Spies [1999], Derpanis [2006], and Sun et al. [2014]. Originally developed for the detection of real world rigid object motion, the technique has been also successfully applied to determine fluid motion from particle and scalar images [Cassisa et al., 2011; Corpetti et al., 2006; Liu and Shen, 2008; Quénot et al., 1998; Ruhnau et al., 2005].

The motion estimation schemes used in this thesis are based on differential methods. Therefore, the principles of some differential methods are recapitulated in the following. An essential requirement of these methods is that the image is differentiable in space and time. Usually, the motion estimation method relies on the assumption of the temporal conservation of an invariant derived from the data [Heitz et al., 2010]. For rigid or quasi-rigid body motion, this can be the geometric invariance of local features such as corners or contours. Because such features are difficult to define for fluid images, photometric invariants are used. This means that a property of the pixels such as the gray value is assumed not to change from one image to the next.

In the next sections differential-based estimation approaches are explained, common constraint equations, also called *data terms* or *observation terms* are introduced, and different solution concepts for the optical flow problem are presented.

3.4.1 Brightness change constraint equation

The most common constraint for image motion estimation is the *brightness change constraint*, which was applied in the seminal paper of Horn and Schunck [1981]. It relates the change of brightness at an image point to the motion of the brightness pattern. The assumption is that the brightness $I(x, y, t)$ of a particular spatio-temporal point in the pattern remains constant. This implies that for a shift of the pattern of the distance δx in x-direction and δy in y-direction during the time δt the brightness constancy model can be formulated as

$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t) \; . \tag{3.3}$$

The right hand side of this equation can be expanded in a Taylor series

$$I(x,y,t) = I(x,y,t) + \delta x\frac{\partial I}{\partial x} + \delta y\frac{\partial I}{\partial y} + \delta t\frac{\partial I}{\partial t} + \mathcal{O}^2 \tag{3.4}$$

where $\mathcal{O}^2$ denotes the second and higher order terms. It is assumed that the spatio-temporal structure around $\boldsymbol{x}$ is locally linear and the higher order terms can be neglected. Simplifying Equation (3.4) by subtracting $I(x,y,t)$ from both sides, ignoring $\mathcal{O}^2$, and dividing by δt yields

$$\frac{\delta x}{\delta t}\frac{\partial I}{\partial x} + \frac{\delta y}{\delta t}\frac{\partial I}{\partial y} + \frac{\partial I}{\partial t} = 0\,. \tag{3.5}$$

By substitution of the optical flow vector $\boldsymbol{u} = (u,v)^T = \left(\frac{\delta x}{\delta t}, \frac{\delta y}{\delta t}\right)^T$ this equation can be reformulated as

$$(\nabla I)^T \cdot \boldsymbol{u} + I_t = 0 \tag{3.6}$$

where ∇I is the 2D vector of spatial gradients of I in x- and y-direction with the 2D gradient operator $\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}\right)^T$. Here and in the following I_z denotes the partial derivative of the image brightness with respect to z and, therefore, I_t is the partial time derivative of I. Equation (3.6) is commonly known as the *brightness change constraint equation* (BCCE). It states that the brightness I of an image at location $\boldsymbol{x}$ can only change due to motion. Strictly speaking, it is only valid for a constant and homogeneous illumination as well as for Lambertian surface properties of the imaged objects. In reality these conditions are rarely met and, therefore, many extensions of the BCCE have been proposed (cf. Section 3.4.3).

3.4.2 Aperture problem

The BCCE (3.6) is an ill-posed equation with the two unknown components u and v of the optical flow $\boldsymbol{u}$. This equation alone only allows to determine the component $\boldsymbol{u}_\perp$ pointing in the direction of the image gradient given by

$$\boldsymbol{u}_\perp = -\frac{I_t \nabla I}{||\nabla I||_2^2} \tag{3.7}$$

where $||\cdot||$ represents the L^2-norm. The direction of $\boldsymbol{u}_\perp$ corresponds to the direction perpendicular to the line of equal intensities. However without further assumptions, the second component of the vector pointing in the direction perpendicular to the image gradient cannot be determined. In the literature this problem is commonly known as *aperture problem* [Beauchemin and Barron, 1995; Ullman, 1979].

The aperture problem can be illustrated by a moving object seen through small apertures as shown in Figure 3.3. By observing the scene through an aperture, only a small part of the object, in this case a square, is visible. Depending on the local structure seen through the aperture, the observer records different motion information. In the first case where the aperture is placed

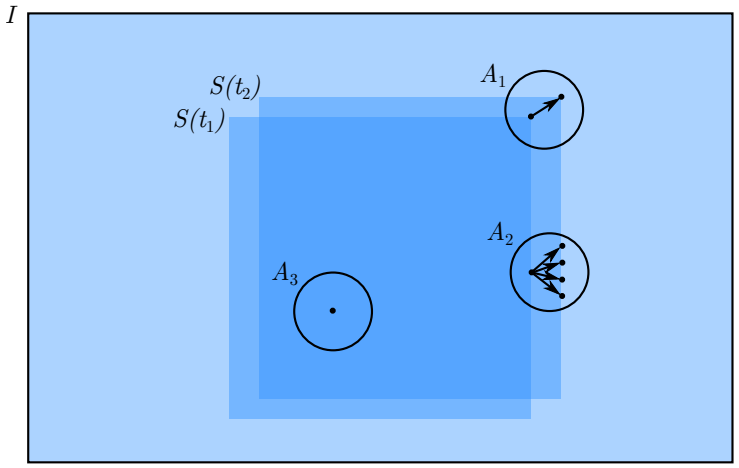

Figure 3.3: Illustration of the *aperture problem*. The blue square is moving from position $S(t_1)$ in the first image taken at time t_1 to position $S(t_2)$ in the second image taken at time t_2. Observing the scene through aperture A_1, the optical flow can be determined correctly because there is enough structure present at the corner of the square. At the edge of the square (aperture A_2) many solutions of the optical flow problem exist and, therefore, the solution is not unique. However, the component $\mathbf{u}_\perp$ in direction of the local gradient, that is, perpendicular to the edge, can be determined. In the center of the homogeneous square at aperture A_3 no information about the local structure is available and the optical flow is zero (blank wall problem).

over the corner of the square (A_1) the correct shift is visible. For the second aperture A_2, only the shift perpendicular to the edge is observable (aperture problem). Through aperture A_3 no change of the structure is observable and, thus, no information about the object motion can be gained and the optical flow is zero. The last case is also known as *blank wall problem* [Derpanis, 2006; Jähne et al., 2007].

3.4.3 Extending the brightness change constraint equation

The conditions, which have to hold for the validity of the BCCE (3.6), are quite strong [Derpanis, 2006; Verri and Tomaso, 1989]. In essence, the underlying surface radiance has to be constant and, therefore, the illumination needs to be uniform. Furthermore, the motion has to be parallel to the sensor, because motion in the direction of the optical axis leads to an intensity change in the image. Also, the surface of the imaged objects has to be well textured and Lambertian, in order to reflect the light equally in all directions. When these conditions are strictly met the optical flow field equals the motion field exactly (cf. Section 3.2). In surface regions with significant irradiance variations due to non-motion effects, the BCCE leads to erroneous results. This is especially the case when these effects lead to gray value changes of the same intensity than gray value changes caused by motion.

To overcome these drawbacks, many extensions of the BCCE have been proposed [Beauchemin and Barron, 1995; Derpanis, 2006; Heitz et al., 2010] , which led to different constraint equations or data models. Cornelius and Kanade [1983] proposed to relax the brightness constancy

assumption by allowing linear brightness changes. Therefore, they introduced a constant term c in the BCCE

$$(\nabla I)^T \cdot \boldsymbol{u} + I_t = c \,. \tag{3.8}$$

Nagel [1989] suggested that the BCCE should be explicitly based on the three dimensional scene. Therefore, knowledge of the scene geometry is required. The constant c is then given by

$$c = 4I \left(\frac{\hat{\boldsymbol{z}} \dot{\boldsymbol{P}}^T}{\hat{\boldsymbol{z}} \boldsymbol{P}^T} - \frac{\boldsymbol{P} \dot{\boldsymbol{P}}^T}{\boldsymbol{P} \boldsymbol{P}^T} \right) \tag{3.9}$$

where $\boldsymbol{P}$ is a three dimensional point in the real world scene, $\dot{\boldsymbol{P}}$ the corresponding velocity and $\hat{\boldsymbol{z}}$ is a unit vector in the direction of the optical axis [Beauchemin and Barron, 1995].

Negahdaripour and Yu [1993] proposed a *generalized brightness change model*, which allows linear transformation of the brightness of an image point, from one instant of time to the next, described by a multiplier and an offset field. The model incorporates brightness variations of scene points, due to non-uniform illumination, light source motion, specular reflection, and interreflection. The general brightness change model extends the brightness constancy model by the multiplier field, which captures brightness changes due to illumination and reflectance changes, and the offset field, which models interreflections, shading variations caused by light source motion, as well as saturation of the sensor due to large variations of the illumination level. The belonging motion constraint equation is given by

$$(\nabla I)^T \cdot \boldsymbol{u} + I_t = m_t I + c_t \tag{3.10}$$

where m_t is the temporal contrast change and c_t the temporal mean intensity change.

Another generalization of the brightness constancy assumption proposed by Haussecker and Fleet [2001] defines a path $\boldsymbol{x}(t)$ along which the brightness can change according to a parameterized function h. The *generalized brightness change constraint equation* is given by

$$(\nabla I)^T \cdot \boldsymbol{u} + I_t = \frac{\mathrm{d}}{\mathrm{d}t} h(I_0, t, \boldsymbol{a}) \tag{3.11}$$

where I_0 is the image at time $t = 0$ and $\boldsymbol{a}$ is a parameter vector of the brightness change model. Brightness changes are either parameterized as time-varying analytical functions or by the differential equations, which model the underlying physical processes. The physical models examined by Haussecker and Fleet [2001] are moving illumination, changing surface orientation, and physical models of heat transport in infrared images such as diffusion or exponential decay.

A further extension of the BCCE can be derived by treating the optical flow like a fluid flow and considering the image brightness to be related to the density of a physical quantity such as the particle concentration in the case of particle images. Then, the original BCCE (3.6) corresponds to the *continuity equation* of incompressible fluids with the condition of the incompressibility given by $\nabla \boldsymbol{u} = 0$, assuming that the continuity equation is also divergence

free in two dimensions. One drawback of this divergence free equation is that it is invalid for brightness changes due to out-of-plane motion. Therefore, Schunck [1984, 1985] proposed to use the continuity equation as data model resulting in a more general constraint equation

$$(\nabla I)^T \cdot \boldsymbol{u} + I \cdot \nabla \boldsymbol{u} + I_t = 0 \tag{3.12}$$

where $\nabla \boldsymbol{u}$ represents the divergence of the flow field. Equation (3.12) is sometimes named *extended optical flow constraint* [Derpanis, 2006]. The constraint was used for instance by Wildes et al. [1997] in the context of experimental fluid mechanics and by Corpetti et al. [2002] for satellite meteorological images. A substantial discussion about the relation between fluid flow and optical flow is given in Liu and Shen [2008].

In the case the brightness constancy assumption fails, the constancy of image features, which are less sensitive to illumination changes can be used. For instance, the gradient of the brightness intensity $\nabla I = (I_x, I_y)^T$ can be assumed to be constant over time [Brox et al., 2004; Bruhn, 2006; Verri et al., 1990]. Because the gradient is a 2D vector one obtains two constraint equations

$$(\nabla I_x) \cdot \boldsymbol{u} + I_{xt} = 0 \tag{3.13}$$

$$(\nabla I_y) \cdot \boldsymbol{u} + I_{yt} = 0\,. \tag{3.14}$$

which can be added up quadratically to generate the data term.

Of course also other features such as higher order derivatives can be considered to be constant and can be used to formulate a data model. Also, linear combinations of different constancy assumptions, e.g., the constancy of brightness and the constancy of the gradient, are feasible [Brox et al., 2004; Bruhn, 2006; Tretiak and Pastor, 1984]. Data terms based on higher order image derivatives are often more sensitive to noise than data terms based on the original brightness data.

The success of each of these constraint equations strongly depends on the frame conditions such as the kind of motion, image acquisition conditions, noise, and prior knowledge of the scene. There is not one single constraint equation which fits best for all optical flow problems.

3.4.4 Differential-based estimation approaches

Differential-based optical flow estimation approaches assume the image sequence to be locally continuous in the spatial and temporal dimensions. In reality, only a discrete representation of the image sequence is available. However, it is assumed that the sampling rate is small in order to recover the derivatives of the underlying continuous representation.

In order to handle the aperture problem described in Section 3.4.2 and determine a solution of the ill-posed BCCE (3.6), several solution concepts can be used. Of course the methods can also be combined with other constrained equations introduced in Section 3.4.3. Depending on the way the optical flow problem is solved, the differential-based estimation schemes can be divided

mainly into local and global approaches.

Local approaches

Since the BCCE (3.6) is an ill-posed problem (cf. Section 3.3), additional constraints are required to achieve a solution. In fact, at least two independent constraint equations are needed to determine the two unknown flow components. The simplest local approach is the *Lucas and Kanade* method [Lucas and Kanade, 1981] where it is assumed that the optical flow is constant in a small neighborhood $\mathcal{N}$ and the information, imposed by one constraint equation per image location, is combined in order to obtain an overdetermined system of equations. From the single BCCE only the flow component $\boldsymbol{u}_\perp$ in direction of the brightness gradient can be computed (cf. Section 3.4.2). By pooling the constraints of small neighborhoods, the complete flow vector can be estimated provided that the neighborhood is sufficiently large to contain enough structure, and the solutions of the single equations are slightly different. Otherwise the constraint equations are not sufficient to solve the aperture problem. However, large neighborhoods cause interpolation of small scale flows and the larger the chosen neighborhood, the more unlikely it is that the velocity is still constant in the whole neighborhood. The task of finding the optimal size of the neighborhood is referred to as the *generalized aperture problem* [Jähne et al., 2007].

The optical flow problem is formulated as minimization problem of the weighted sum of the squared constraint equation (3.6). It is defined by the normal equation

$$\boldsymbol{u} = \arg\min \sum_{\boldsymbol{x}\in\mathcal{N}} w(\boldsymbol{x}) \left((\nabla I)^T \cdot \boldsymbol{u} + I_t\right)^2 \tag{3.15}$$

where $w(\boldsymbol{x})$ is a weighting function. In order to solve Equation (3.15), the method of *least squares* (cf. Section 2.3.3) is used. The searched solution is the velocity vector, which minimizes the sum of squared deviations of the brightness constancy for each point within the neighborhood $\mathcal{N}$. The least squares solution is given by

$$\boldsymbol{u} = \left(A^T A\right)^{-1} A^T \boldsymbol{b} \tag{3.16}$$

with matrix $A^T A$ and vector $A^T \boldsymbol{b}$ denoted by

$$A^T A = \sum_{\boldsymbol{x}\in\mathcal{N}} w(\boldsymbol{x}) \begin{pmatrix} I_x I_x & I_x I_y \\ I_y I_x & I_y I_y \end{pmatrix} \qquad \text{and} \qquad A^T \boldsymbol{b} = -\sum_{\boldsymbol{x}\in\mathcal{N}} w(\boldsymbol{x}) \begin{pmatrix} I_x I_t \\ I_y I_t \end{pmatrix},$$

respectively. If the inverse of matrix $A^T A$ does not exist, no flow vector can be determined. In this case only the flow $\boldsymbol{u}_\perp$ in direction of the brightness gradient can be estimated.

By applying the Lucas and Kanade approach it is assumed that only the temporal derivatives of the image intensity are corrupted by errors such as Gaussian noise. The spatial derivatives, however, are considered to be noise free. A generalization is the *structure tensor approach*, where it is assumed that both, spatial and temporal derivatives are contaminated by noise [Bigün et al.,

1991; Nagel and Gehrke, 1998]. Instead of using the least squares solution, within the structure tensor approach the method of *total least squares* (cf. Section 2.3.3) is utilized to solve the optical flow problem (3.6). Similar to the Lucas and Kanade approach, the structure tensor approach can be formulated as minimization problem

$$\tilde{\boldsymbol{u}} = \underset{\tilde{\boldsymbol{u}}^T\tilde{\boldsymbol{u}}=1}{\arg\min} \sum_{\boldsymbol{x}\in\mathcal{N}} w(\boldsymbol{x}) \left((\nabla_3 I)^T \cdot \tilde{\boldsymbol{u}}\right)^2 \tag{3.17}$$

where $w(\boldsymbol{x})$ is a weighting function and $\nabla_3 I$ is the three dimensional vector with two spatial and one temporal derivative and $\tilde{\boldsymbol{u}} = (u, v, 1)^T$ is an extended optical flow vector. In order to avoid the trivial solution $\tilde{\boldsymbol{u}} = 0$, an additional constraint $\tilde{\boldsymbol{u}}^T\tilde{\boldsymbol{u}} = 1$ is applied. Equation (3.17) can be converted into

$$\tilde{\boldsymbol{u}} = \underset{\tilde{\boldsymbol{u}}^T\tilde{\boldsymbol{u}}=1}{\arg\min} = \tilde{\boldsymbol{u}}^T J \tilde{\boldsymbol{u}} \tag{3.18}$$

with the *structure tensor* J, which is defined by

$$J := \sum_{\boldsymbol{x}\in\mathcal{N}} w(\boldsymbol{x}) \begin{pmatrix} I_x I_x & I_x I_y & I_x I_t \\ I_y I_x & I_y I_y & I_y I_t \\ I_t I_x & I_t I_y & I_t I_t \end{pmatrix}. \tag{3.19}$$

Equation (3.18) can be solved with the help of Lagrange multipliers yielding the eigenvalue problem

$$J\tilde{\boldsymbol{u}} = \lambda\tilde{\boldsymbol{u}}\ . \tag{3.20}$$

The minimum of Equation (3.18) is the eigenvector $\tilde{\boldsymbol{u}}$ corresponding to the smallest eigenvalue λ of the structure tensor J. The optical flow vector $\boldsymbol{u}$ is then given by the first two components of the normalized eigenvector. Thereby, the eigenvector is normalized in a way that the third component equals one.

The structure tensor is widely used in image processing to estimate local orientations. Thereby, the orientation at a point is determined with local gradients considering a small local neighborhood around this point. Inclusion of the information of the neighboring points is usually achieved by a convolution with a Gaussian kernel K_σ with standard deviation σ (cf. Section 2.3.4). The weighted sum over the neighborhood $\mathcal{N}$ in Equation (3.19) is, therefore, expressed by the convolution with K_σ, where σ is related to the size of $\mathcal{N}$. Considering the structure tensor, optical flow estimation can also be interpreted as the finding of the spatio-temporal orientation with the least change of the gray value which is given by the eigenvector of the structure tensor to the smallest eigenvalue.

The Lucas and Kanade and the structure tensor approach share the assumption that the optical flow is constant in small local neighborhoods. According to Brox [2005], ordinary least squares tends to underestimate the flow vectors at motion discontinuities, whereas, total least squares often yields estimates which are too large, especially if the orientation was estimated

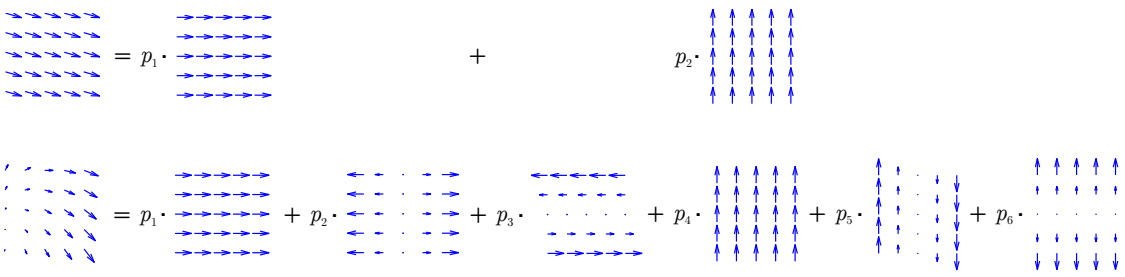

Figure 3.4: Illustration of parametric methods as a linear sum of orthogonal basis flows. *Top row*: Lucas and Kanade method with two parameters and flow models of constant flow. *Bottom row*: Parametric method with six parameters and affine motion models.

wrong.

The local approaches discussed so far are special cases of the *parametric approach* where the optical flow is modeled in small local neighborhoods by a parametric function [Derpanis, 2006]. The simplest function is the one of constant flow and accords to the methods described above. More complex models have to be applied in a way that spatial variations occurring within the local neighborhoods, can be described by the model. Instead of directly estimating the flow vector, specific parameters need to be estimated, which implicitly define the flow vector as the center of a small local neighborhood of vectors given by a linear combination of the parameters and the motion models.

One example of a parametric approach with more advanced flow models uses *affine motion models*. It assumes linear variation of the optical flow by an affine transformation of local image regions [Haussecker and Spies, 1999; Szeliski, 2006]:

$$\boldsymbol{u} = \begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_1 \\ t_2 \end{pmatrix} = A\boldsymbol{x} + \boldsymbol{t} \; . \tag{3.21}$$

The model superimposes affine motion. The vector $t = (t_1, t_2)^T$ represents translation motion. It corresponds to the constant optical flow vector $\boldsymbol{u}$ used in the structure tensor and the Lucas and Kanade method. From the parameters of the affine transformation matrix A the basic elementary geometric transformations such as rotation, dilation, shear and stretching can be determined.

In order to estimate the optical flow, the flow vector $\boldsymbol{u}$ in Equation (3.6) is replaced by the affine motion model or by any other parametric model obtaining a similar quadratic optimization problem as before:

$$\arg\min \sum_{\boldsymbol{x} \in \mathcal{N}} w(\boldsymbol{x}) \left((A\boldsymbol{x} + \boldsymbol{t})^T (\nabla I) + I_t \right)^2 \; . \tag{3.22}$$

Minimization can be done once again by the method of least squares to solve for the parameters of the model. Inserting the parameters in the underlying model (e.g., Equation (3.21)) yields the flow vector.

An illustration of the parametric method is shown in Figure 3.4. Within the Lucas and Kanade method only two models of constant flow are applied. The affine approach utilizes six parameters and affine motion models. The flow vector $\boldsymbol{u}$ is then given by a linear combination of motion models and parameters p. The neighborhood reconstructed with the two parameter model contains the same vector at every location, whereas the flow vectors estimated with the six parameter model differ from each other. By using more than two parameters it is possible to model local variations within the neighborhood. The optical flow method introduced later on in this thesis (cf. Section 4.1.3) is a parametric method with a user-defined number of parameters and learned motion models.

Velocity fields estimated with local methods are sometimes not dense. This is due to the fact that the optical flow cannot be determined in areas with no or only little structure. In these areas, the matrix of the least squares estimation becomes singular and no flow vector can be estimated, resulting in a sparse flow field. In order to state the reliability of single flow vectors, *confidence measures* can be used. One example for a confidence measure is to observe the smallest eigenvalue of the least squares matrix, which is an indicator of how close the matrix is to being singular [Derpanis, 2006; Simoncelli et al., 1991]. If the eigenvalue is close to zero, the matrix is not invertible and only the normal flow component $\boldsymbol{u}_\perp$ can be calculated.

Global approaches

Global approaches form another class of methods to solve the underconstrained flow problem. They are also called variational approaches. The most prominent global approach, which is also the basic framework of other variational methods, was introduced by Horn and Schunck [1981]. In order to solve the BCCE (3.6), they proposed the additional use of a global smoothness constraint, which is to minimize the squared magnitude of the gradient of the optical flow components given by

$$||\nabla u||_2^2 \qquad \text{and} \qquad ||\nabla v||_2^2 \,. \tag{3.23}$$

By using this smoothness constraint, one assumes that neighboring image points have similar velocities and, thus, the velocity field of the brightness pattern varies smoothly almost everywhere. In order to make the optical flow problem well-posed, the solution is regularized by using the so-called *Tikhonov* regularizer [Tikhonov and Arsenin, 1977]. The combination of the BCCE with the regularizer based on the smoothness assumption yields the energy functional

$$E = \int_\Omega \left((\nabla I)^T \cdot \boldsymbol{u} + I_t\right)^2 + \lambda \left(||\nabla u||_2^2 + ||\nabla v||_2^2\right) \mathrm{d}\boldsymbol{x} \tag{3.24}$$

with the positive regularization parameter λ with which the influence of the smoothness constraint can be controlled by the user. Larger values of λ lead to a stronger penalization of large gradients and, therefore, yield smoother flow fields. The functional is defined over the image

domain Ω. The first part is often called *data term* whereas the second term is called *smoothness* or *regularization term*.

In order to estimate the optical flow f the energy of (3.24) has to be minimized. This can be done due to *calculus of variations* (cf. Section 2.3.1) by solving the *Euler-Lagrange* equations. The smoothness term thereby restricts the class of admissible solutions. The Euler-Lagrange equations are given by

$$I_x(I_x u + I_y v + I_t) - \lambda \Delta u = 0 \tag{3.25}$$

$$I_y(I_x u + I_y v + I_t) - \lambda \Delta v = 0 \tag{3.26}$$

with the *Laplace operator* $\Delta = (\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2})$, which can be approximated numerically using finite differences [Horn and Schunck, 1981]. Therefore, it is $\Delta u(x, y) = \bar{u}(x, y) - u(x, y)$ with the weighted average $\bar{u}(x, y)$ calculated from u in the neighborhood of location (x, y). Of course Δv is approximated similarly.

The system of equations (3.25) and (3.26) can be solved iteratively for the two components u and v of the optical flow vector $\boldsymbol{u}$ by

$$u^{(q+1)} = \bar{u}^{(q)} - \frac{I_x(I_x \bar{u}^{(q)} + I_y \bar{v}^{(q)} + I_t)}{\lambda + I_x^2 + I_y^2} \tag{3.27}$$

$$v^{(q+1)} = \bar{v}^{(q)} - \frac{I_y(I_x \bar{u}^{(q)} + I_y \bar{v}^{(q)} + I_t)}{\lambda + I_x^2 + I_y^2} \ . \tag{3.28}$$

Thereby, the subscript $q + 1$ denotes the next iteration step.

An advantage of global methods such as the Horn and Schunck method is that it yields dense flow fields as flow information is interpolated from the surrounding to areas with homogeneous brightness pattern and no or only little structure information. At image locations with small gradients the data term of (3.24) is close to zero and the influence of the smoothness term is large. Since the total energy is minimized the smoothness term gets small as well, which favors flow values that are similar to the neighboring values. This is called *filling-in effect* [Bruhn et al., 2005].

However, global methods are more prone to noise than local methods [Barron et al., 1994]. Since noise leads to relatively strong gradients, which serve as weights in the data term, the influence of the smoothness term is weak compared to that of the data term at noisy locations. Therefore, the flow field is less regularized at noisy image structures than elsewhere. The regularization parameter λ has to be chosen carefully to obtain good filling-in properties on the one hand, and low sensitivity to noise on the other hand. A large λ leads to potentially good looking flow fields which may be oversmoothed and therefore false.

In general, global methods are more accurate than local methods [Barron et al., 1994]. However, local methods are usually faster and simpler to implement. Another advantage of local methods is that they are more robust to noise than global methods because in local methods

local errors do not propagate to the surrounding estimates as it is the case in global methods.

An extension of the Horn and Schunck approach in presence of occluding boundaries was described by Nagel and Enkelmann [1986]. They used an *image-driven* smoothness constraint. Therefore, they replaced the standard smoothness constraint by an oriented smoothness term that preserves edges by not smoothing over them. The edges are identified by steep intensity gradients. However, steep gradients do not necessarily belong to edges, they also appear for instance in strong textured regions of an object. In such a case a smoothing over the edge would be desirable but is indeed prevented. Alternatively so-called *flow-driven* constraints (cf. Section 3.4.5), which reduce smoothing across flow discontinuities can be applied [Weickert and Schnörr, 2001].

Instead of a purely spatial, also spatio-temporal smoothness constraints can be applied [Black and Anandan, 1991; Bruhn et al., 2005]. Due to the additional denoising properties in temporal direction spatio-temporal constraints may lead to better results.

Corpetti et al. [2006] showed that applying the Horn and Schunck smoothness constraint is equal to using the so-called *first-order div-curl regularizer* given by

$$\int_{\Omega} (\mathrm{div}\boldsymbol{u})^2 + (\mathrm{curl}\boldsymbol{u})^2 \tag{3.29}$$

with $\mathrm{div}\boldsymbol{u} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}$ and $\mathrm{curl}\boldsymbol{u} = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$. This formulation shows that the regularization term may not be well suited for fluid flows, since such flows are expected to contain vorticity and divergence, which are in fact suppressed by the regularization. In turbulent flows for example there are many rotational structures, which are characterized by a large vorticity. To overcome this problem, *higher-order regularizers*, which penalize the gradient of the divergence and the curl of the flow, can be used [Corpetti et al., 2002; Yuan et al., 2007]. For instance the *second-order div-curl regularizer* was proposed by Suter [1994] and is given by

$$\int_{\Omega} ||\nabla \mathrm{div}\boldsymbol{u}||_2^2 + ||\nabla \mathrm{curl}\boldsymbol{u}||_2^2 \,. \tag{3.30}$$

Using the second-order div-curl regularizer (3.30) the divergence and the vorticity of the flow field is preserved.

Combined approaches

Both, local and global methods, have advantages and disadvantages. It seems natural to combine the two techniques in order to get a new combined approach featuring the best of both worlds. The most famous combined approach is the one introduced by Bruhn et al. [2005], where the local Lucas and Kanade method is combined with the global Horn and Schunck method. The advantage of a combined approach is the robustness under Gaussian noise while giving dense flow fields.

Applying the augmented optical flow vector $\tilde{\boldsymbol{u}} = (u, v, 1)^T$ and the structure tensor J defined

in Equation (3.19) the energy functional can be formulated analogously to the functional (3.24) of the Horn and Schunck method:

$$E = \int_{\Omega} \tilde{\boldsymbol{u}}^T J \tilde{\boldsymbol{u}} + \lambda \left(||\nabla u||_2^2 + ||\nabla v||_2^2 \right) \mathrm{d}\boldsymbol{x} \; . \tag{3.31}$$

The data term described in the first part of the functional contains the local Lucas and Kanade method in structure tensor notation. Additionally, as regularization term, embodied in the second part of the functional, the same smoothness constraint already utilized within the Horn and Schunck method is used. In order to find the optimal optical flow vector, which is the one that minimizes (3.31), the Euler-Lagrange equations have to be solved. They are given by

$$J_{11} u + J_{12} v + J_{13} - \lambda \Delta u = 0 \tag{3.32}$$

$$J_{12} u + J_{22} v + J_{23} - \lambda \Delta v = 0 \; . \tag{3.33}$$

A solution of these linear equations can be obtained with iterative methods such as for instance SOR introduced in Section 2.3.2.

Extensions of the method use robust error norms (cf. Section 3.4.5) instead of quadratic ones or are integrated into a spatio-temporal framework [Bruhn et al., 2005]. One can also apply different local models as data term such as for instance the structure tensor model or the learning-based model, which will be introduced in Section 4.1.4.

Physics-based optical flow estimation

The regularizations used so far to achieve a solution for the optical flow problem are not motivated by physical concepts. For example, the smoothness assumption (3.23) is a merely reasonable assumption which is needed to make the problem well posed. In the context of fluid dynamics, the fluid flow is described by the *Navier-Stokes* equations. It seems natural, to use the knowledge that the fluid flow has to satisfy these flow equations and utilize them as physics-based constraint equations. Therefore, it is desirable to develop a computational scheme, which consistently combines the evaluation of experimental data and simulations [Heitz et al., 2010].

A first step in this direction was made by Nakajima et al. [2003]. They proposed an energy functional for incompressible fluids which uses the continuity equation and the Navier-Stokes equations as regularization. The functional over the image domain Ω can be written as

$$E = \int_{\Omega} \lambda_1 \left((\nabla I)^T \cdot \boldsymbol{u} + I_t \right)^2 + (\nabla \boldsymbol{u})^2 + \lambda_2 \left(\frac{\mathrm{D}\boldsymbol{u}}{\mathrm{D}t} + \frac{1}{\rho} \nabla p - \nu \Delta \boldsymbol{u} - \boldsymbol{f} \right)^2 \mathrm{d}\boldsymbol{x} \tag{3.34}$$

where the differential operator $\frac{\mathrm{D}\boldsymbol{u}}{\mathrm{D}t}$ is given by $\left(\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} \right)$. Δ is the Laplace operator, ρ the density of the fluid, p the pressure, ν the kinetic viscosity and $\boldsymbol{f}$ the external forces on a unit mass of fluid. The coefficients λ_1 and λ_2 are used to weight the terms differently.

The flow field is estimated by solving the corresponding non-linear Euler-Lagrange equations.

Since the result of non-linear optimization problems often depends on the initial condition, Nakajima et al. [2003] propose to use the estimated flow field of Horn and Schunck's method as initial condition in order to permit a stable calculation. They then apply several computation steps with gradually reduced weight parameter λ_1. The obtained solution is a combination of the solution of the data term and the solution of the fluid flow equations.

Another approach combining image-based flow measurements with the constitutive fluid flow equations was proposed by Ruhnau and Schnörr [2007]. In a first attempt they restrict the approach to steady flows of viscous media. Therefore, all admissible flow fields have to fulfill the *Stokes* equations, which are the underlying physical equations for this kind of problems. The restriction to the Stokes equation is done mainly for simplicity reasons as the problem gets very complex for general flows described by the Navier-Stokes equation. The approach is formulated according to other variational approaches and comprises the objective functional

$$E(\boldsymbol{u}, p, \boldsymbol{f}, \boldsymbol{g}) = \int_\Omega \left((\nabla I)^T \cdot \boldsymbol{u} + I_t\right)^2 + \alpha ||\boldsymbol{f}||_2^2 \,\mathrm{d}\boldsymbol{x} + \int_{\partial\Omega} \gamma ||\nabla_t \boldsymbol{g}||_2^2 \,\mathrm{d}s \tag{3.35}$$

and a system of physical constraints denoted by the Stokes equations and boundary conditions

$$\begin{cases} -\mu\Delta\boldsymbol{u} + \nabla p &= \boldsymbol{f} \qquad \text{in}\,\Omega\,, \\ \nabla\boldsymbol{u} &= 0 \qquad \text{in}\,\Omega\,, \\ \boldsymbol{u} &= \boldsymbol{g} \qquad \text{on}\,\partial\Omega\,. \end{cases} \tag{3.36}$$

Again, p denotes the pressure, $\boldsymbol{f}$ the body force acting on a unit mass of fluid, μ the dynamic viscosity, and $\boldsymbol{g}$ the boundary values defined on the boundary $\partial\Omega$ of the domain Ω. The term $\nabla_t \boldsymbol{g}$ denotes the derivative of $\boldsymbol{g}$ tangential to the boundary. To make the whole problem well posed, additional regularization terms of the control variables weighted with coefficients α and γ are included into the objective functional.

In order to estimate the optical flow field, the functional (3.35) is minimized subject to Equation (3.36). The principle of the method, is that body force $\boldsymbol{f}$ and boundary values $\boldsymbol{g}$ are determined in a way that the resulting velocity field $\boldsymbol{u}$ matches the apparent motion given by the data term (3.6) as accurate as possible. In terms of control theory $\boldsymbol{f}$ and $\boldsymbol{g}$ are called control variables. In the case of the velocity field being *de facto* governed by the Stokes equations the pressure p and the forces $\boldsymbol{f}$ can directly be determined from the image sequence. However, the approach can also be applied successfully to divergence-free turbulent flows, which cannot be described by the Stokes equations. For these flows, the control variables $\boldsymbol{f}$ and $\boldsymbol{g}$ are no longer physically significant but still control the flow [Heitz et al., 2010]. In their work Ruhnau and Schnörr [2007] showed that the approach can outperform other optical flow methods as well as correlation-based methods in case of highly non-rigid flows.

A limitation of the approach is that the boundary conditions must be known in order to estimate the flow field. Yet, a complete knowledge of the boundary is unavailable in many fluid

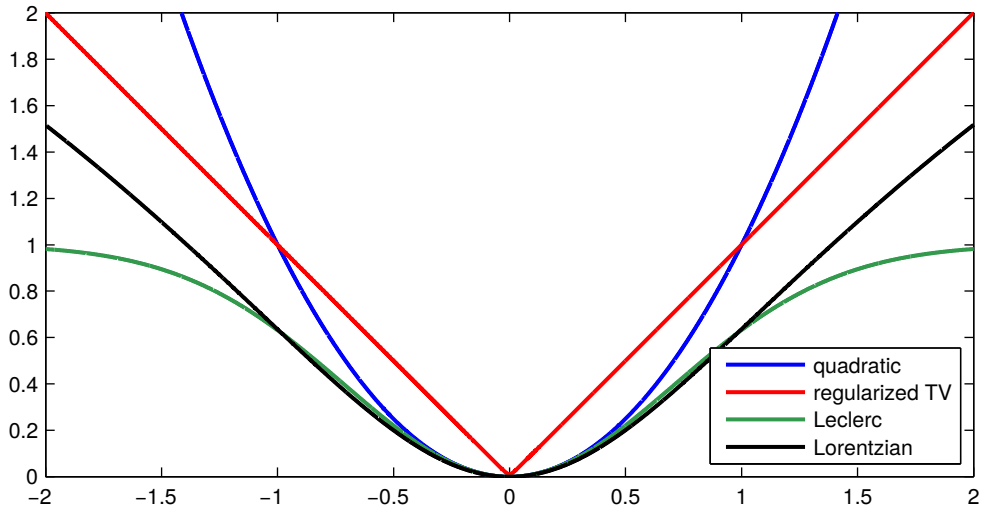

Figure 3.5: Different error norms, including the quadratic norm, the convex *regularized TV* penalty function ($\epsilon = 0.01$), and the two non-convex error norms *Leclerc* ($\tau = 1$) and *Lorentzian* ($\sigma = 0.75$).

mechanical applications. Due to the complexity of the approach it is computationally quite heavy and also detailed expertise is required in order to solve the underlying optimal control problem and implement the approach.

3.4.5 Robust error functions

Within the original variational approach proposed by Horn and Schunck [1981] quadratic error functions are used for the data and the regularization term, respectively. This is a good choice in the case of the occurring errors being Gaussian as well as independent and identically distributed [Baker et al., 2011]. However, in the case of outliers originating from illumination changes, occlusion, or noise it seems beneficial to penalize these outliers less severely than in a quadratic way. A remedy may be to use so-called *robust penalty functions*, which originate from robust statistics [Huber, 1981], for the data term. By such a function the influence of outliers on the BCCE is reduced.

Nevertheless, the use of such a robust error function is also reasonable for the regularization term, since the quadratic error norm is prone to outliers, which do not fulfill the smoothness assumption. With a robust error function the regularizer can handle discontinuities in the optical flow field. A regularizer that applies a non-quadratic error norm, is called *flow-driven* [Weickert and Schnörr, 2001]. In contrast, there are also *image-driven* regularizers that associate discontinuities of the image sequence with discontinuities in the flow field [Nagel and Enkelmann, 1986; Weickert and Schnörr, 2001].

The standard optical flow energy functional (cf. Equation (3.24)) with robust error functions is given by

$$E = \int_\Omega \psi_d \left((\nabla I)^T \cdot \boldsymbol{u} + I_t \right) + \lambda \psi_s \left(||\nabla u||_2 + ||\nabla v||_2 \right) \mathrm{d}\boldsymbol{x} \; . \tag{3.37}$$

Thereby, the non-quadratic penalizer functions $\psi_d(\cdot)$ and $\psi_s(\cdot)$ are used for the data term and the smoothness term, respectively. This robust energy functional accepts outliers in the data term as well as in the smoothness term.

Many robust error norms have been proposed for optical flow applications [Black and Anandan, 1996; Stewart, 1999; Sun et al., 2014] some of which are depicted in Figure 3.5. The regularized *total variation* (TV) penalty function $\psi(x) = \sqrt{x^2 + \epsilon^2}$ with the small positive constant ϵ is a convex error function which corresponds to a regularized L^1-norm [Bruhn et al., 2005]. Convex error norms have the advantage that they guarantee well-posedness and convergence of the problem because they have a unique minimum. This means that the solution is independent from the initialization. There are also non-convex error norms such as the *Leclerc* penalty function $\psi(x) = 1 - \exp(-\tau x^2)$ [Corpetti et al., 2006] or the *Lorentzian* penalty function $\psi(x) = \log\left(1 + \frac{x^2}{2\sigma^2}\right)$ [Sun et al., 2014] with the positive parameters τ and σ, respectively. From a statistical point of view, these non-convex penalizers are more robust but the resulting energy functional may comprise multiple local minimums and, therefore, advanced minimization strategies have to be applied in order to find the global minimum.

Local optical flow techniques can also be transformed into a robust formulation using the error norm $\psi(\cdot)$ [Fleet and Weiss, 2006]. A robust version of Equation (3.15) thus reads

$$\boldsymbol{u} = \arg\min \sum_{\boldsymbol{x} \in \mathcal{N}} w(\boldsymbol{x}) \psi\left((\nabla I)^T \cdot \boldsymbol{u} + I_t\right) \ . \tag{3.38}$$

Another robust local approach used in the field of motion detection with data corrupted by non-Gaussian noise is the *least median of squares* method. The approach is given by solving the minimization problem [Derpanis, 2006]

$$\boldsymbol{u} = \arg\min \mathrm{med_i}\left[\left((\nabla I)^T \cdot \boldsymbol{u} + I_t\right)_i^2\right] \ . \tag{3.39}$$

In the case of Gaussian noise, the efficiency of the least median of squares method is poor. An extension of the method was proposed by Bab-Hadiashar and Suter [1998]. They used the least median of squares to find an initial estimate, which is used to assign each equation to either the group of inliers or the group of outliers. The solution is given by applying the method of least squares to the group of inliers.

3.4.6 Hierarchical multi-scale methods

Optical flow estimation schemes are often realized as hierarchical multi-scale approaches [Bruhn, 2006; Mémin and Pérez, 1998, 2002]. These approaches are either applied to speed up the estimation process or to improve the results. Especially the variational formulation is limited to small displacements compared to the shortest wavelength present in the image [Heitz et al., 2010]. Displacements that are larger than half of the period of the highest frequency component can cause temporal aliasing and are a potential source of local minima, in which the estimation

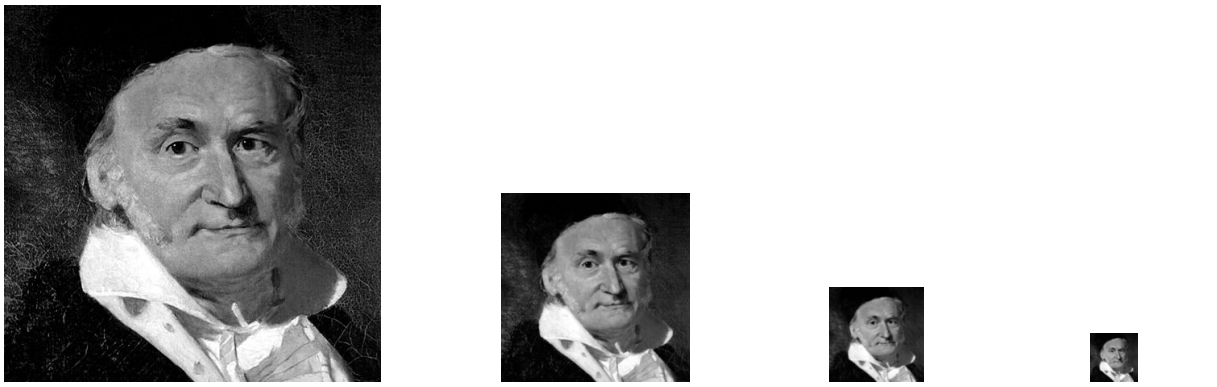

Figure 3.6: Example of a Gauss pyramid on a black and white image of Carl Friedrich Gauss (painting by Gottlieb Biermann, 1887). The resolution decreases from left to right.

can get trapped [Derpanis, 2006]. Furthermore, the linearization of the BCCE (Equation (3.4)) also restricts the optical flow estimation to relatively small displacements. In order to handle large displacements the estimation process is performed step wise in a coarse to fine manner. Therefore, the images of the sequence are transformed into image pyramids such as for instance the Gauss pyramid.

The Gauss pyramid consists of multiple versions of an image, which all have a different resolution and size. Stacking these different versions on each other with the next smaller image on top of the previous one, a pyramid like structure is obtained. The Gauss pyramid is created successively by applying a smoothing filter followed by a sub-sampling step to the image of the previous scale [Jähne, 2005]. The smoothing filter is necessary to follow the *sampling theorem*, which states that a periodic structure is only sampled correctly, if at least two samples per wavelength are taken [Jähne, 2005]. This means that the size reduction always has to go together with an adequate smoothing operation. An example of a Gauss pyramid is shown in Figure 3.6. To perform the smoothing operation the image is convolved with a Gaussian kernel K_σ of standard deviation σ as introduced in Section 2.3.4. After the convolution the image is down sampled with the sampling rate $\downarrow r$. In this example, a sampling rate of $\downarrow r = 2$ is used, which means that the dimension of the image on the next pyramid level is cut in half. The pyramid is built step wise since image $I^{(q)}$ on the q-th level of the pyramid is calculated from image $I^{(q-1)}$ on the $(q-1)$-th level. Therefore, the smoothing and down sampling operations are done successively on the subsequent images.

The hierarchical multi-scale scheme applied within this thesis combines the estimated optical flow on each pyramid level with an image warping step. The optical flow field estimated at a coarser pyramid level is used to warp the original image. Therefore, the flow field is up-sampled to fit the dimension of the current pyramid level, and the warped image is an interpolated image obtained from the original image shifted by this flow field. At the next finer level, the remaining motion between the original and the warped image is estimated. Finally, a summation of the

motion increments of all levels yields the desired optical flow field.

This hierarchical coarse to fine technique improves the quality of the estimated optical flow in cases of large displacements. However, a drawback of the multi-resolution approach is that it sometimes produces wrong estimates when large estimation errors occur on coarser scales that cannot be corrected on the finer scales [Derpanis, 2006]. Another limitation is that the technique restricts the entire estimation process more or less to image pairs. The use of temporal information by the application of image sequences within the coarse to fine technique, requires a large number of successive warping steps for the very same image, which makes the approach very complex and introduces large interpolation errors.

An alternative multi-resolution technique, which can be used in connection with variational optical flow techniques and which requires no warping steps, is to use the flow field estimated on a coarser scale as initial guess on the next finer scale [Anandan, 1989]. This technique is on the one hand applied to improve the resulting flow field and on the other hand to speed up the calculation due to a faster convergence of the solution.

3.5 Error measures

Different *performance* or *error* measures can be used to quantitatively evaluate the accuracy of motion estimation algorithms. Accordingly, the estimated motion field is compared to the correct flow field. Of course this comparison is only possible if the correct flow data is known. Otherwise, if no reference flow field is available the second image of a sequence can be compared to a warped version of the first image obtained by shifting the gray values by the estimated motion field. In the following section the error measures applied in this thesis are described.

3.5.1 Angular error

The *angular error* (AE) is the most popular performance measure of optical flow applications in computer vision [Baker et al., 2011] but it is also used in the field of fluid dynamics [Héas et al., 2013; Ruhnau et al., 2005]. Following Barron et al. [1994] the AE is defined as

$$\mathrm{AE} := \arccos\left(\frac{\tilde{\boldsymbol{u}}_c \tilde{\boldsymbol{u}}_e}{\|\tilde{\boldsymbol{u}}_c\| \, \|\tilde{\boldsymbol{u}}_e\|}\right) \cdot \frac{180}{\pi} \tag{3.40}$$

using the extended 3D vectors $\tilde{\boldsymbol{u}}_c = (u_c, v_c, 1)$ for the correct and $\tilde{\boldsymbol{u}}_e = (u_e, v_e, 1)$ for the estimated flow field. These vectors can be interpreted as orientation vectors in the 3D spatio-temporal domain (cf. Figure 3.7) with time difference $t = 1$ between successive images. The AE gives the angular deviation from the correct spatio-temporal orientation vector in units of degree.

The popularity of the AE comes from the fact that it takes into account both, direction and magnitude differences between the vector fields. Another advantage of the AE compared to other error measures is that it avoids the division by zero for zero flows and is, therefore, well-defined.

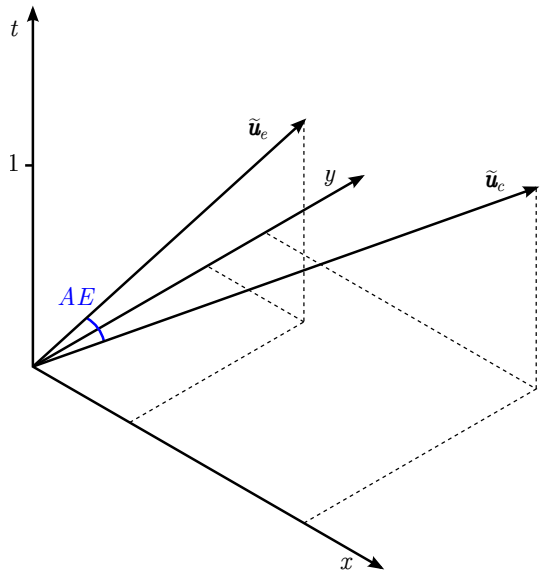

Figure 3.7: Illustration of the *angular error* (AE). The angel in the 3D spatio-temporal space between the correct flow vector $\tilde{\mathbf{u}}_c$ and the estimated flow vector $\tilde{\mathbf{u}}_e$ is defined as AE.

It is also able to handle large and small displacements without the amplification inherent in relative measures of vector differences.

However, the AE has some limitations as shown by Haussecker and Spies [1999]. It depends for instance non-linearly on the speed of the velocity field. Also, symmetric deviations from the true velocity do not lead to the same AE. Thus, a velocity vector estimated 10% shorter than the true vector has a different AE than a velocity vector estimated 10% longer [Nieuwenhuis, 2009].

The mean value of the AE of the entire flow field is defined as *average angular error* (AAE). This is a single scalar value, which describes the overall deviation of the estimated flow field from the correct one and is a convenient way to quantify the performance, thereby ranking different motion estimators.

3.5.2 Displacement error

An alternative error measure, which is often used for PIV image sequences [Stanislas et al., 2008] is the *average displacement error* (ADE). It is defined as the *root mean square* (RMS) of the deviation of the estimated displacements from the correct displacements for the x-component and the y-component, respectively

$$\mathrm{ADE_x} = \sqrt{\frac{\sum_{i=1}^{m}\sum_{j=1}^{n}\left(u_c(i,j)-u_e(i,j)\right)^2}{mn}} \tag{3.41}$$

$$\mathrm{ADE_y} = \sqrt{\frac{\sum_{i=1}^{m}\sum_{j=1}^{n}\left(v_c(i,j)-v_e(i,j)\right)^2}{mn}}\,. \tag{3.42}$$

The variables m and n denote the width and the height of the flow field, respectively. The ADE is given in units of pixels.

3.5.3 Interpolation error

If the correct velocity field is not available, the AAE and the ADE cannot be assigned. In this case the *average interpolation error* (AIE), which is given by the RMS of the gray value differences of the real and an interpolated image, can be used as error measure. Hereby, the interpolated image is obtained by shifting or *warping* the first image of an image pair by the estimated displacement field towards the second image. According to Baker et al. [2011], the AIE is defined as

$$\mathrm{AIE} = \sqrt{\frac{\sum_{i=1}^{m} \sum_{j=1}^{n} \left(I^{t+1}(i,j) - I_w^t(i,j)\right)^2}{mn}} \tag{3.43}$$

where I^t denotes an image of size $m \times n$ of the sequence at time t and I_w^t denotes a warped version of the same image. To generate I_w^t the estimated velocity field is used to warp the image I^t onto I^{t+1}. A limitation of such error measures is the strong dependence on the interpolation method. Within this thesis a *bicubic* interpolation was applied. Another limitation of the AIE is that in homogeneous image regions with the same gray values, incorrect flow vectors may be considered correct since the gray value difference is vanishing.

4 The learning-based approach

The *learning-based approach* (LBA) is a new method for the estimation of fluid dynamical displacement fields from particle images. The approach is an extension of the model-based optical flow algorithm proposed by Nieuwenhuis et al. [2010] and is based on the approaches introduced by Black et al. [1997] and by Yacoob and Davis [1998], where learned motion models are incorporated into a local optical flow method. The basic idea of the approach is that complex, spatio-temporal flow field patches can be expressed as a linear combination of typical basis flows, which have been learned previously from appropriate training data. In this way, prior knowledge is introduced to the system and the resulting flow field is restricted to the space spanned by these basis flows. The coefficients of the linear combination can be estimated using the brightness constancy assumption described in Section 3.4.1. Some of the basic principles of learning domain specific motion models originate from face recognition where dimension reduction algorithms are used to determine effective subspaces for the representation and discrimination of faces [Li and Jain, 2011]. Especially the concepts of the *eigenface* method introduced by Turk and Pentland [1991] are used for the learning process in an adapted way to meet the requirements of the fluid dynamical framework.

The description of the LBA is organized as follows. Section 4.1 states the principles of the approach and describes the learning process of the basis flows, which are also called motion models. Thereafter, different estimation methods applying the learned motion models are introduced in order to determine the optical flow field. In Section 4.2 some implementation details are described. Finally, the properties and the performance of the approach is described in Section 4.3.

4.1 Description of the approach

4.1.1 Principles

The LBA is an extension of optical flow methods which assume constant flow in local areas such as the local structure tensor approach by Bigün et al. [1991] or the combined local global method by Bruhn et al. [2005]. Both methods are described in Section 3.4.4. Instead of using local neighborhoods of constant flow, typical flow constellations specially designed to meet the requirements of the underlying flow problem, are applied to describe the flow in the neighborhoods. These typical flow structures are embodied by the so-called *motion models* which are learned from a set of training vector fields. Using the motion models, the flow can be described

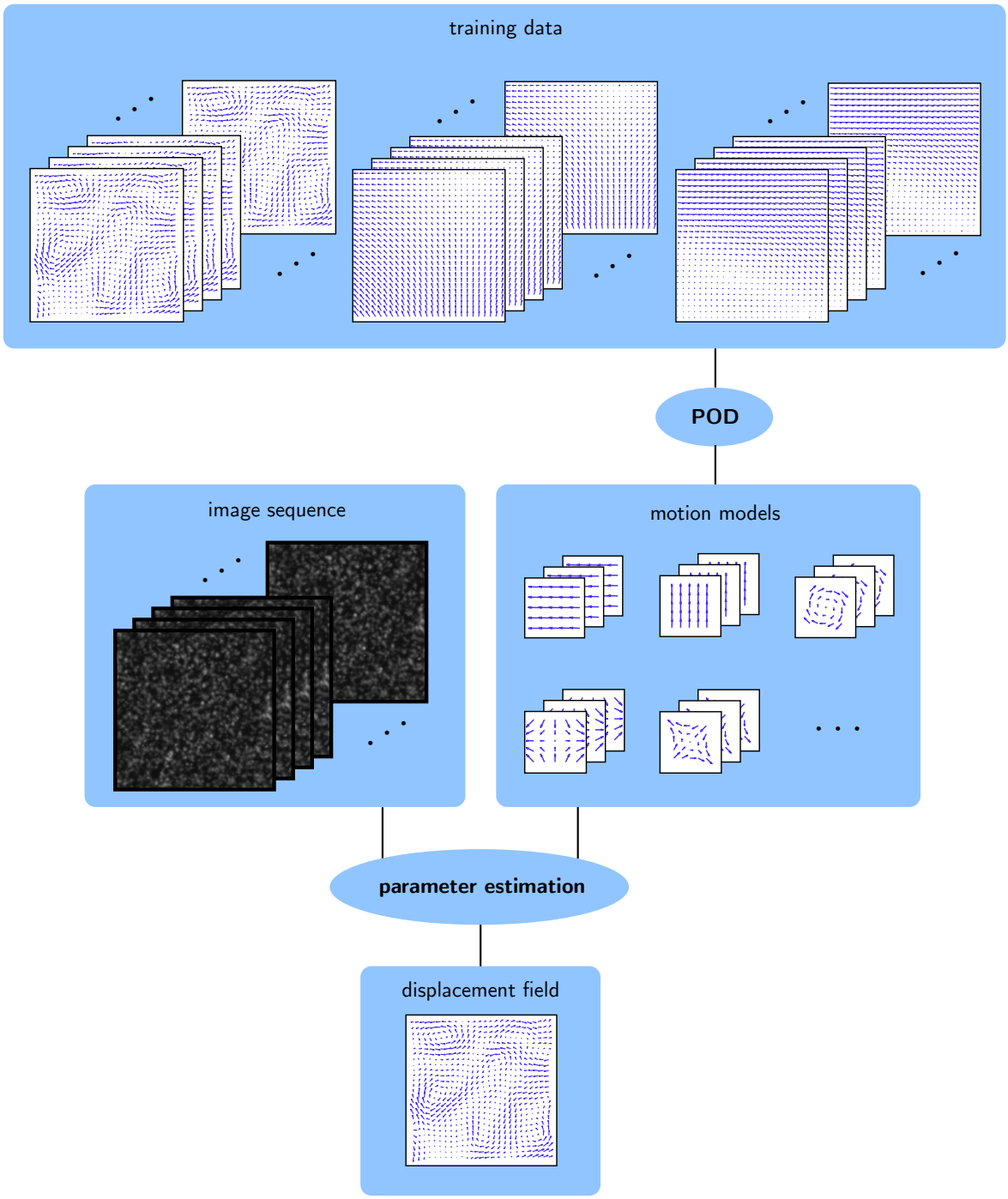

Figure 4.1: Sketch of the learning-based optical flow approach. Initially, typical motion models (here of size $5 \times 5 \times 3$) are learned from the training sequences with POD. In order to estimate the velocity field from the image sequence, a parameter vector has to be determined. A linear combination of these parameters and the motion models yields the velocity field.

more precisely in the local neighborhoods than by the model of constant flow, since occurring local flow structures are considered. Another advantage of the motion models is that a larger model size can be used without over-smoothing allowing for a more accurate estimation due to a larger number of motion constraints.

Since the LBA is a parametric approach, the flow vectors are not directly determined from the optical flow equations. Instead a large number of motion constraints is pooled in a neighborhood to estimate a much smaller number of model parameters. This leads to accurate and stable results. The flow vectors are given by a linear combination of the learned motion models and the parameters. The LBA is similar to the affine model approach described in Section 3.4.4 which uses motion models that consist of affine transformations [Fleet and Weiss, 2006; Haussecker and Spies, 1999]. One disadvantage of the affine model is the limited applicability to complex natural scenes including motion discontinuities or non-rigid motion [Black et al., 1997]. However, the learned motion models may include such complex flow structures if they are present in the training data. In that way, the motion models are tunable for different problems. Fleet et al. [2000] proposed to use an extended affine model, which consists of the affine basis flows and additional learned motion models that are orthogonal to the affine set.

The input ensemble from which the motion models are learned consists of spatio-temporal flow field patches obtained from the training data. Different sets of simulated flow fields or arbitrary flow fields of similar flow applications can be used as training data. By choice of the training data the method can easily be adapted and optimized for different flow situations. If the flow field under study contains for instance rotations, the training data must comprise rotations as well. Otherwise rotations are not included in the motion models and thus, have to be approximated by the other remaining models, which is less accurate. Therefore, excluding rotations from the training data results in an incomplete model.

The basic principles of the LBA are sketched in Figure 4.1. At first, the spatio-temporal motion models are learned from the training data by applying POD (cf. Section 4.1.2). As explained in Section 2.2, POD is used in fluid dynamics to recover the most energetic structures and dominant flow events of turbulent flow fields. A further feature of POD is the ability to reconstruct incomplete, erroneous flow fields. Essentially, POD yields a set of optimal, orthogonal basis functions spanning the space of possible flow structures. Therefore, a reasonable choice is to use the first K most energetic POD modes as motion models.

Secondly, some parameters have to be estimated from the image sequence and the motion models. For this estimation, different local, global, and robust approaches are at hand.

Finally the resulting flow vector $\boldsymbol{u}$ is given in dependance of the local spatio-temporal neighborhood ω by a linear combination of the vectorized motion models $\boldsymbol{\phi}_k$ and the estimated parameters α_k:

$$\boldsymbol{u}(\omega) \approx \sum_{k=1}^{K} \alpha_k \boldsymbol{\phi}_k \; . \tag{4.1}$$

As shown in Figure 4.2, the linear combination of the estimated parameters and the learned

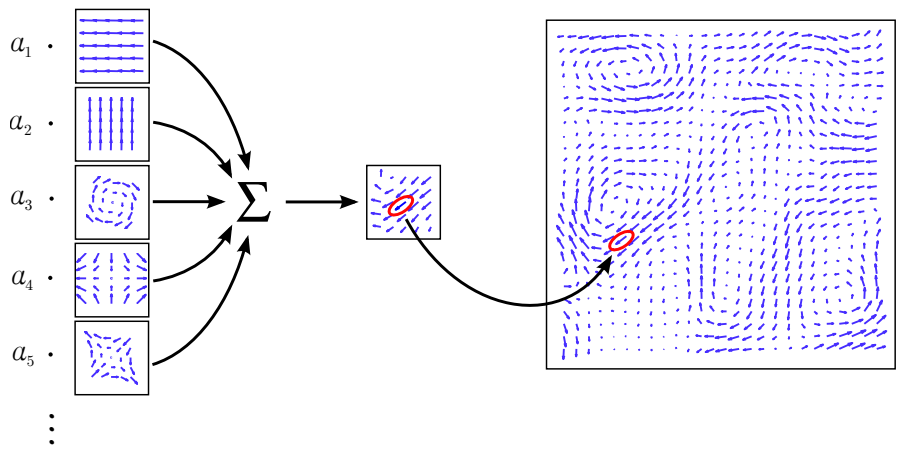

Figure 4.2: Determination of the flow vector. A linear combination of the estimated coefficients α and learned motion models produces a neighborhood of flow vectors of the same size as the basis flows. The central vector of the determined patch is taken as one vector of the resulting flow field.

motion models yields a local neighborhood ω of flow vectors of the same size as the motion models. The central flow vector of this neighborhood is used as flow estimate in the resulting flow field. In this way the flow vectors are determined in dependance of the motion models and the solution is restricted to the space spanned by these motion models. In order to obtain the full vector field, Equation (4.1) has to be solved for every image location.

4.1.2 Learning typical motion models

Consider a training vector field $\boldsymbol{q}(x, y, t)$ defined on the discrete spatio-temporal domain $\mathcal{D} \in \mathbb{R}^3$ where $(x, y) \in \mathbb{R}^2$ indicates the spatial location within a rectangular domain and $t \in [0, T]$ indicates time. To learn the motion models, a defined number of several thousand (e.g., 5000) spatio-temporal patches of fixed size $\omega \in \mathcal{D}$ (e.g., $15 \times 15 \times 7$) are randomly chosen from $\boldsymbol{q}$ as shown in Figure 4.3. Accordingly, the patches may overlap but it is not allowed to choose the exact same patch twice. The number of sample patches must be large enough to ensure that all relevant flow structures are represented according to their significance within the set of sample patches. The influence of the number of sample patches on the resulting motion models was observed. In most cases 5000 patches were sufficient to capture the relevant flow structures and a higher number of patches did not alter the resulting motion models.

In order to prevent any directional bias of the motion models, the sample patches are rotated by 90°, 180°, and 270° and the flow vectors in the patches are mirrored on the horizontal and the vertical axis and moreover, the temporal direction of the flow field in the patches is reversed. By applying these transformations, the motion models become more general and can be used for a broader range of applications. However, in some cases it may be beneficial to omit the transformations in order to obtain motion models that are more specific for a particular application. The influence of the patch transformations on the resulting motion

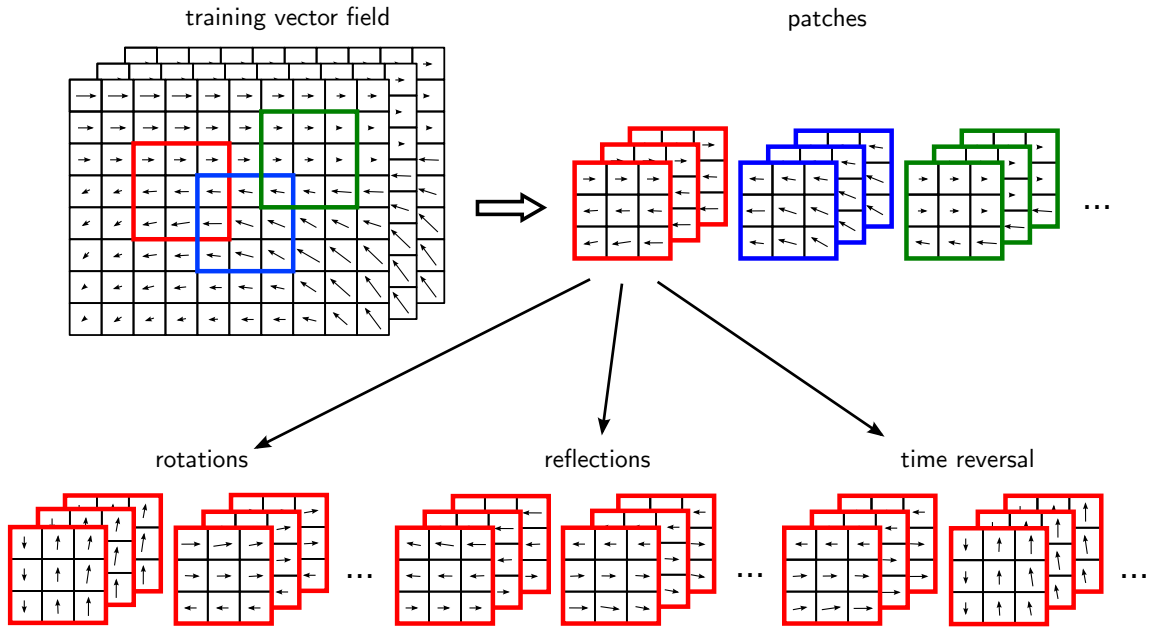

Figure 4.3: The sample patches are chosen randomly from the training vector field $\mathbf{q}(x, y, t)$. In this example the patch size is $3 \times 3 \times 3$. All of the patches are rotated and mirrored and also the time is reversed by changing the sign.

models is described in Section 5.2.1.

In order to apply a POD on the sample patches they have to be transformed into a data matrix, which contains the patches as column vectors. This is done by writing the entries of each patch in lexicographical order in one column vector. Accordingly, the horizontal velocity components are stored in the upper half of the vector and the vertical components in the lower half. The procedure is shown in Figure 4.4. Finally, a large data matrix A of zero mean, which contains the different realizations of the vectorized patches in its columns, is obtained. A is of size $2N \times M$ with N being the number of flow vectors in one patch. The factor two is due to the fact that every flow vector has two components. M indicates the number of patches times the rotations, reflections and time reversals of the patches. Since the spatio-temporal patches are of limited size the number of components in one patch is usually smaller than the number of patches in total, thus, $2N < M$.

The large number of sample patches has to be reduced to only a few motion models, which however, contain most of the information of the entire set of patches. To fulfill this task, the POD is used, which yields a set of optimal, orthonormal basis functions that are sorted, due to their relevance, as shown in Section 2.2. Therefore, applying a POD on matrix A yields a set of $2N$ orthogonal basis functions ϕ_k with $k \in [1, 2N]$. The POD is performed by means of a SVD as described in Section 2.2.3 given by

$$A = U\Sigma V^T \tag{4.2}$$

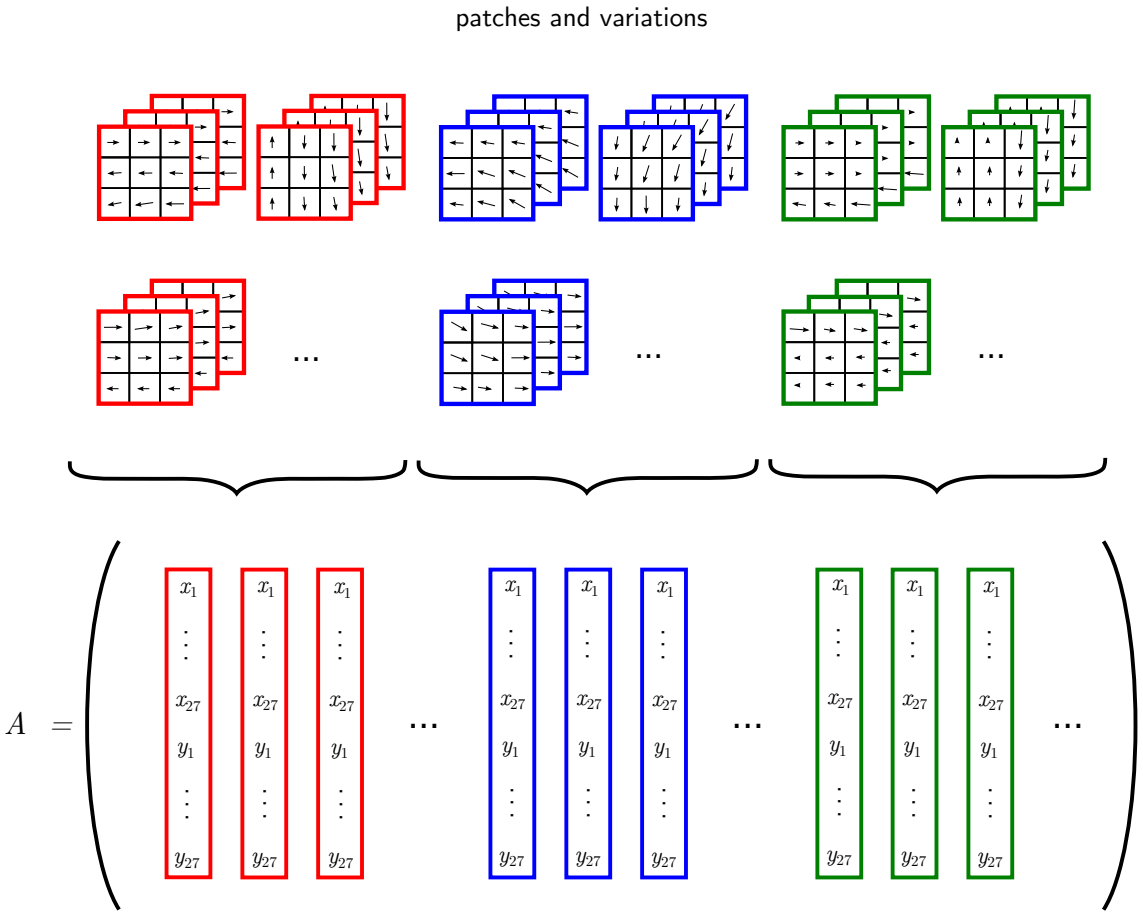

Figure 4.4: Construction of the data matrix. The sample patches and their variations are vectorized and stored as column vectors in the data matrix A.

with the orthogonal $2N \times 2N$ matrix U and the orthogonal $M \times M$ matrix V. The $2N \times M$ diagonal matrix Σ contains the $2N$ singular values in decreasing order $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_{2N} \geq 0$. The column vectors of U define the basis functions, which are also known as POD modes. In practice the SVD is computed in an economic manner to save computation time. Therefore, redundant information is dropped and only the first $2N$ columns of V and Σ are determined. Figure 4.5 shows the matrices of the SVD. The column vectors of U are reshaped back to patch size in order to obtain the motion models.

The motion models are sorted by their information content and, therefore, the set of motion models can be cropped nearly without loss of information. As shown in Figure 2.3 in Section 2.2, the singular values and, thus, the information content drop quite fast, which implies that the first motion models are sufficient to approximate any flow vector. According to Section 4.1.1 the approximation of the flow vector is given by the linear combination (4.1) of motion models ϕ_k and motion parameters α_k. The motion models implicitly encode all the relevant information of the fluid flow. The ability of the POD modes alias motion models to approximate any state of a

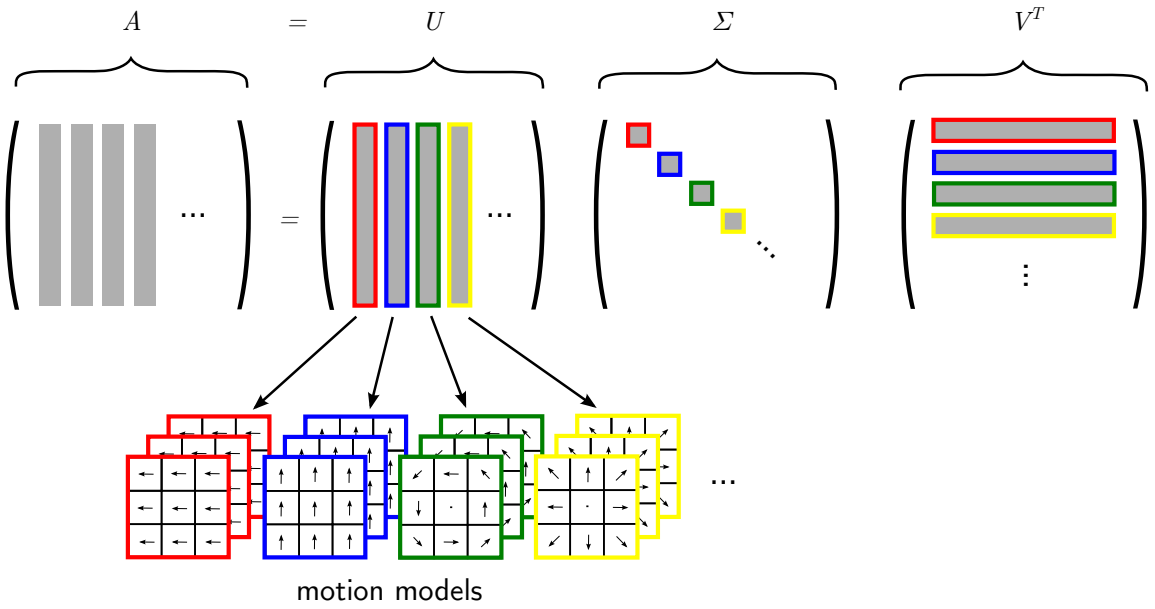

Figure 4.5: Visualization of the SVD. The motion models are given by the reshaped column vectors of U.

complex flow system is totally dependent on the information originally contained in the patches obtained from the training data. Therefore, the training data has to be extensive to include all relevant flow structures.

The following sections describe how the parameters α_k are estimated from the image sequence in dependence of the motion models ϕ_k. Essentially, the estimation is done either via a local parametric technique or a combined local global approach.

4.1.3 Local approach

The estimation of the parameters can be performed in a local manner similar to the method proposed by Lucas and Kanade [1981] and is therefore called *local learning-based approach* (lLBA). The Lucas and Kanade method utilizes only two parameters and two motion models of constant flow in x- and y-direction, whereas, the number of parameters and motion models within the lLBA can be chosen arbitrarily. As part ot this thesis the lLBA has already been published by Stapf and Garbe [2013, 2014a] and is based on the work of Black et al. [1997] and Nieuwenhuis et al. [2010].

The BCCE introduced in Section 3.4.1, which assumes constancy of the image brightness I is used as data term. The BCCE is given by

$$(\nabla I)^T \cdot \boldsymbol{u} + I_t = 0 \,. \tag{4.3}$$

It is assumed that the 2D optical flow vector $\boldsymbol{u}$ can be approximated in dependance of a local neighborhood ω by the linear combination of Equation (4.1). A substitution of $\boldsymbol{u}$ within (4.3)

by (4.1) leads to a system of N linear equations

$$\widetilde{\nabla}\boldsymbol{I} \cdot \sum_{k=1}^{K} \alpha_k \boldsymbol{\phi}_k = -\boldsymbol{I}_t \tag{4.4}$$

with the N-dimensional vector of gray values $\boldsymbol{I}$ which can be understood as a vectorized local neighborhood ω of gray values of the same size as the learned motion models $\boldsymbol{\phi}_k$. Then $\boldsymbol{I}_t$ is defined as the vector of partial derivatives of the gray value at each position within the local neighborhood with respect to time. And $\widetilde{\nabla}\boldsymbol{I}$ is defined by the $N \times 2N$ gradient matrix:

$$\widetilde{\nabla}\boldsymbol{I} := \begin{pmatrix} I_{x1} & 0 & \cdots & 0 & I_{y1} & 0 & \cdots & 0 \\ 0 & I_{x2} & 0 & \cdots & 0 & I_{y2} & & 0 \\ \vdots & & \ddots & & \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & I_{xN} & 0 & \cdots & 0 & I_{yN} \end{pmatrix} .$$

Here, I_{xi} denotes the gradient in x-direction of the i-th patch position and I_{yi} denotes the gradient in y-direction of the i-th patch position. Together with the $2N \times K$ matrix Φ, which contains the first K motion models $\phi_1, \cdots, \phi_K$ as column vectors, $\widetilde{\nabla}\boldsymbol{I}$ can be combined to the $N \times K$ matrix B, it is $B := \widetilde{\nabla}\boldsymbol{I} \cdot \Phi$. Using B the system of linear equations (4.4) can be simplified to

$$B \cdot \boldsymbol{\alpha} = -\boldsymbol{I}_t \tag{4.5}$$

with the K-dimensional parameter vector $\boldsymbol{\alpha}$. In order to determine the parameter vector $\boldsymbol{\alpha}$, this system of linear equations can be solved by the method of least squares as introduced in Section 2.3.3. Therefore, the solution is given by

$$\boldsymbol{\alpha} = -\left(B^T B\right)^{-1} B^T \boldsymbol{I}_t \, . \tag{4.6}$$

As with all local optical flow methods, the estimated displacement field may not be dense because Equation (4.6) has no solution in cases where $B^T B$ is not invertible. This can happen for instance in homogeneous image regions with very little structure or in areas where the aperture problem (cf. Section 3.4.2) is present. However, by using the *Moore-Penrose pseudoinverse* [Ben-Israel and Greville, 2003] a solution of the problem can be obtained. In order to identify locations where the determination of the optical flow is problematic, confidence measures such as the one based on linear subspace projections proposed by Kondermann et al. [2007] can be applied. This confidence measure assumes, that all correct flow field patches can be described by a linear combination of the learned motion models. This means, that the better the flow constellations can be reconstructed in terms of the motion models, the more reliable they are.

4.1.4 Variational approach

The *variational learning-based approach* (vLBA) is an extension of the lLBA into a global framework. Analog to the local-global optical flow approach proposed by Bruhn et al. [2005] (cf. Section 3.4.4) the local approach is thereby embedded into a global functional. This alternative formulation of the problem combines advantages of local and global optical flow methods, such as robustness against noise and dense flow fields, respectively. In locations with no or only little structure, local methods may fail to determine an optical flow vector. Global methods, however, fill in information from the surrounding and are able to state flow vectors in these areas. Additionally, advantages of parametric optical flow approaches are contained within the vLBA. Using more advanced motion models than constant flow and more than two parameters, allows to precisely model the flow within small local neighborhoods and leads to better flow estimates. As part of this thesis the approach has already been published [Stapf and Garbe, 2014b].

As data term basically the system of linear equations (4.4) is used. Additionally, a further constraint in form of a regularizing term is included within the functional. Similar to the regularizer proposed by Horn and Schunck [1981], which assumes smooth variation of the optical flow and penalizes large gradients of the optical flow vector in a quadratic way, the squared magnitude of the gradient of the parameters given by

$$\|\nabla \alpha_k\|_2^2 \,, \qquad k = 1, \ldots, K \tag{4.7}$$

was chosen as regularizer. Thus, global smoothness of the parameters is assumed and the gradient of the parameters is quadratically penalized. This simplest case of regularization leads to linear equations and is, therefore, relatively easy to solve. Using the following notations

$$\begin{aligned} \tilde{\boldsymbol{\alpha}} &:= (\alpha_1, \ldots, \alpha_K, 1)^T \\ \boldsymbol{l}_k &:= \widetilde{\nabla} \boldsymbol{I} \cdot \boldsymbol{\phi}_k \\ L &:= (\boldsymbol{l}_1, \ldots, \boldsymbol{l}_K, \boldsymbol{I}_t) \\ J &:= L^T L \\ \|\nabla \boldsymbol{\alpha}\|_2^2 &:= \sum_{k=1}^{K} \|\nabla \alpha_k\|_2^2 \end{aligned} \tag{4.8}$$

the energy functional can be expressed by

$$E = \int_\Omega \tilde{\boldsymbol{\alpha}}^T J \tilde{\boldsymbol{\alpha}} + \lambda \|\nabla \boldsymbol{\alpha}\|_2^2 \, \mathrm{d}\boldsymbol{x} \tag{4.9}$$

where the first part denotes the data term and the second part denotes the regularizing term. The ratio of the two terms is defined by the positive regularization parameter λ, which can be set by the user. The optimal parameter vector $\boldsymbol{\alpha}$ is the one that minimizes this functional. Due to the principles of *calculus of variations* the minimum of a functional can be obtained by solving

the *Euler-Lagrange* equations (cf. Section 2.3.1) given by a set of K equations:

$$\sum_{j=1}^{K} J_{ij}\alpha_j + J_{iK+1} - \lambda\Delta\alpha_i = 0\,, \qquad i = 1,\ldots,K \tag{4.10}$$

with the subscripts $i, j \in [1, K]$ and the Laplace operator Δ defined by $\Delta := \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$.

Due to the discrete nature of the input images the continuous Euler-Lagrange equations have to be discretized. Therefore, a rectangular pixel grid of cell size h_x and h_y is applied. In general, image processing uses fixed grids with $h = h_x = h_y = 1$. All values are given at discrete locations $(m, n) \in \mathbb{N}^2$ with distance h between neighboring points. The number of pixels in x-direction and in y-direction is denoted by N_x and N_y, respectively. Let $\mathcal{N}(m, n)$ denote the set of the four nearest neighbors of pixel (m, n), then the 2nd order central difference approximation at location (m, n) of $\Delta\alpha_i$ is given by

$$\Delta\alpha_i \approx \sum_{\tilde{m},\tilde{n}\in\mathcal{N}(m,n)} \frac{\alpha_{\tilde{m}\tilde{n},i} - \alpha_{mn,i}}{h^2}\,. \tag{4.11}$$

Using the above approximations and notations the Euler-Lagrange equations (4.10) can be discretized. At location (m, n) they are given by

$$\frac{1}{\lambda}\left(\sum_{j=1}^{K} J_{mn,ij}\alpha_{mn,j} + J_{mn,iK+1}\right) - \sum_{\tilde{m},\tilde{n}\in\mathcal{N}(m,n)} \frac{\alpha_{\tilde{m}\tilde{n},i} - \alpha_{mn,i}}{h^2} = 0\,, \qquad i = 1,\ldots,K\,. \tag{4.12}$$

For $m = 1,\ldots,N_y$ and $n = 1,\ldots,N_x$ this is a large sparse system of linear equations and like other variational optical flow methods, it can be solved iteratively. According to Bruhn et al. [2005], the SOR method, introduced in Section 2.3.2, is a good compromise between simplicity and efficiency for combined local and global optical flow approaches and is, therefore, used in the vLBA as well. In the following the iteration parameter q and the notations

$$\begin{aligned}\mathcal{N}^-(m,n) &:= \{\tilde{m},\tilde{n} \in \mathcal{N}(m,n) \mid \tilde{m} < m, \tilde{n} < n\}\\ \mathcal{N}^+(m,n) &:= \{\tilde{m},\tilde{n} \in \mathcal{N}(m,n) \mid \tilde{m} > m, \tilde{n} > n\}\end{aligned} \tag{4.13}$$

are used. The iterative SOR solution of the discrete Euler-Lagrange equations is given by

$$\begin{aligned}\alpha_{mn,i}^{(q+1)} = {}& (1-\omega)\,\alpha_{mn,i}^{(q)}\\ &+ \omega\,\frac{\sum\limits_{\tilde{m},\tilde{n}\in\mathcal{N}^-(m,n)} \alpha_{\tilde{m},\tilde{n},i}^{(q+1)} + \sum\limits_{\tilde{m},\tilde{n}\in\mathcal{N}^+(m,n)} \alpha_{\tilde{m},\tilde{n},i}^{(q)}}{|\mathcal{N}(m,n)| + \frac{h^2}{\lambda}J_{mn,ii}}\\ &- \omega\,\frac{\frac{h^2}{\lambda}\left(\sum\limits_{j=1,j<i}^{K} J_{mn,ij}\alpha_{mn,j}^{(q+1)} + \sum\limits_{j=1,j>i}^{K} J_{mn,ij}\alpha_{mn,j}^{(q)} + J_{mn,iK+1}\right)}{|\mathcal{N}(m,n)| + \frac{h^2}{\lambda}J_{mn,ii}}\end{aligned} \tag{4.14}$$

for $i = 1, \ldots, K$, $m = 1, \ldots, N_y$ and $n = 1, \ldots, N_x$. $|\mathcal{N}(m,n)|$ denotes the number of the next neighbors of pixel (m,n), which belong to the image domain. The value of the relaxation parameter ω is chosen between 1 and 2 in order to speed up the convergence compared to the Gauss-Seidel method. Initially $\boldsymbol{\alpha}^{(0)}$ is assigned by 0.

Notable when spatio-temporal motion models are utilized the data term of (4.9) contains in addition to spatial also temporal information. To extend the regularization term to the spatio-temporal domain as well, the 2D Laplace operator in Equation (4.9) must be replaced by a 3D spatio-temporal one. This leads to an enlarged set of nearest neighbors $\mathcal{N}(m,n)$, which also contains the nearest neighbors in temporal direction.

The discrete Euler-Lagrange equations can also be written in matrix notation. Accordingly, a large sparse matrix is required. In order to simplify the matrix notation, smaller matrices are stitched together. To achieve this, the following notations are used:

$$s_{mn,ij} := \frac{h^2}{\lambda} J_{mn,ij} + 4$$
$$t_{mn,ij} := \frac{h^2}{\lambda} J_{mn,ij}$$

$$S_{ij} := \begin{pmatrix}
s_{11,ij} & -1 & 0 & \cdots & \cdots & 0 & -1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\
-1 & s_{12,ij} & -1 & \ddots & & & \ddots & -1 & \ddots & & & & \vdots \\
0 & -1 & s_{13,ij} & -1 & \ddots & & & \ddots & \ddots & \ddots & & & \vdots \\
\vdots & \ddots & -1 & \ddots & \ddots & \ddots & & & \ddots & \ddots & \ddots & & \vdots \\
\vdots & & \ddots & \ddots & \ddots & -1 & \ddots & & & \ddots & \ddots & \ddots & \vdots \\
0 & & & \ddots & -1 & s_{1N_x,ij} & -1 & \ddots & & & \ddots & \ddots & 0 \\
-1 & \ddots & & & \ddots & -1 & s_{21,ij} & -1 & \ddots & & & \ddots & -1 \\
0 & \ddots & \ddots & & & \ddots & -1 & s_{22,ij} & -1 & \ddots & & & 0 \\
\vdots & \ddots & \ddots & \ddots & & & \ddots & -1 & \ddots & \ddots & \ddots & & \vdots \\
\vdots & & \ddots & \ddots & \ddots & & & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
\vdots & & & \ddots & \ddots & \ddots & & & \ddots & \ddots & \ddots & \ddots & 0 \\
\vdots & & & & \ddots & \ddots & \ddots & & & \ddots & \ddots & \ddots & -1 \\
0 & \cdots & \cdots & \cdots & \cdots & 0 & -1 & 0 & \cdots & \cdots & 0 & -1 & s_{N_yN_x,ij}
\end{pmatrix}$$

$$T_{ij} := \begin{pmatrix}
t_{11,ij} & 0 & \cdots & 0 \\
0 & t_{12,ij} & \ddots & \vdots \\
\vdots & \ddots & \ddots & 0 \\
0 & \cdots & 0 & t_{N_yN_x,ij}
\end{pmatrix}$$

On the diagonal of the matrix S_{ij} the values of J_{ij} at location (m,n) multiplied with $\frac{h^2}{\lambda}$ are

written together with the central pixel of the 2nd order central difference term. The four off-diagonals of S_{ij} contain the four next neighbors of the central pixel. The diagonal of T_{ij} contains the values of J_{ij} at location (m, n) multiplied with $\frac{h^2}{\lambda}$. With these notations the large sparse matrix can be written as:

$$B := \begin{pmatrix} S_{11} & T_{12} & \cdots & \cdots & T_{1K} \\ T_{21} & S_{22} & \ddots & & T_{2K} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & T_{K-1K} \\ T_{K1} & \cdots & \cdots & T_{KK-1} & S_{KK} \end{pmatrix} \tag{4.15}$$

The discrete Euler-Lagrage equations in matrix notation are then given by the sparse system of linear equations

$$B\boldsymbol{\alpha} = \boldsymbol{b} \ . \tag{4.16}$$

Hereby, the parameter vector $\boldsymbol{\alpha}$ with stacked parameter values at each image location and each motion model and the right hand side vector $\boldsymbol{b}$ are given by:

$$\begin{aligned} \boldsymbol{\alpha} &:= \left(\alpha_{11,1} \dots \alpha_{N_y N_x,1}, \alpha_{11,2} \dots \alpha_{N_y N_x,2}, \dots, \alpha_{11,K} \dots \alpha_{N_y N_x,K}\right)^T \\ \boldsymbol{b} &:= \left(J_{11,1K+1} \dots J_{N_y N_x,1K+1}, J_{11,2K+1} \dots J_{N_y N_x,2K+1}, \dots, J_{11,KK+1} \dots \alpha_{N_y N_x,KK+1}\right)^T . \end{aligned} \tag{4.17}$$

Using the decomposition $B = D+L+U$ with diagonal matrix D, lower triangular matrix L, and upper triangular matrix U the SOR solution of the Euler-Lagrange equations in matrix notation is given by (cf. Equation (2.24))

$$\boldsymbol{\alpha}^{(q+1)} = (D + \omega L)^{-1} \left(\omega \boldsymbol{b} + ((1-\omega) D - \omega U) \boldsymbol{\alpha}^{(q)}\right) \ . \tag{4.18}$$

Unless otherwise specified, the vLBA is conducted with a maximum number of 50 iteration steps and a SOR parameter $\omega = 1.8$. However, if a fixed point is reached in less than 50 iteration steps, the calculation is stopped successfully.

4.1.5 Robust variational approach

So far the vLBA uses a quadratic error norm for the data and the regularization term. This error norm has the advantage that the underlying energy functional results in linear Euler-Lagrange equations which are relatively easy to solve. However, the use of a quadratic L^2 norm implies, that the occurring errors are assumed to be Gaussian as well as independent and identically distributed [Baker et al., 2011]. In reality, however, this assumption is frequently violated as for instance near occlusion boundaries. Therefore, robust error norms introduced in Section 3.4.5 may yield better results. Robustness implies that the method is less sensitive to the influence of outliers in the input data [Bab-Hadiashar and Suter, 1998]. The *robust variational learning-*

based approach (rvLBA) introduced in this section, uses the robust error functions $\psi_d(\cdot)$ and $\psi_s(\cdot)$ for the data term and the smoothness term, respectively.

One example of a robust error norm is the L^1 norm, which falls into the class of TV [Weickert and Schnörr, 2001] known from noise reduction [Rudin et al., 1992]. It is beneficial to use a convex error norm, because this guarantees that the problem has a unique minimizer. The regularized version of the TV penalty function given by

$$\psi(x^2) = \sqrt{x^2 + \epsilon^2} \tag{4.19}$$

with the regularization parameter ϵ, is therefore a good choice, because it allows rather sharp discontinuities [Brox, 2005]. For simplicity reasons, this regularized error function is used within the rvLBA for the data term and for the regularization term.

With the notations given in Equation (4.8) the energy functional can be expressed similar to (4.9) with robust error norms by:

$$E = \int_\Omega \psi_d\left(\tilde{\boldsymbol{\alpha}}^T J \tilde{\boldsymbol{\alpha}}\right) + \psi_s\left(\lambda \|\nabla \boldsymbol{\alpha}\|_2^2\right) \mathrm{d}\boldsymbol{x} \ . \tag{4.20}$$

In order to determine the minimizing parameter vector $\boldsymbol{\alpha}$, the Euler-Lagrange equations have to be solved. They are given by:

$$\psi_d'\left(\tilde{\boldsymbol{\alpha}}^T J \tilde{\boldsymbol{\alpha}}\right)\left(\sum_{j=1}^{K} J_{ij}\alpha_j + J_{iK+1}\right) - \lambda \operatorname{div}\left(\psi_s'\left(\|\nabla \boldsymbol{\alpha}\|_2^2\right)\nabla \alpha_i\right) = 0 \ , \qquad i = 1, \ldots, K \ . \tag{4.21}$$

Accordingly, ψ_d' and ψ_s' denote the first derivative of each error function. Because in both cases Equation (4.19) is used as error function, the derivative is given by

$$\psi'(x^2) = \frac{1}{2\sqrt{x^2 + \epsilon^2}} \tag{4.22}$$

in each case. Both terms of Equation (4.21) are non-linear because of the non-linear factors $\psi_d'\left(\tilde{\boldsymbol{\alpha}}^T J \tilde{\boldsymbol{\alpha}}\right)$ and $\psi_s'\left(\|\nabla \boldsymbol{\alpha}\|_2^2\right)$ in front of the linear expressions known from (4.10). Due to this non-linearity the solution of (4.21) is slightly more complicated. Similar to the vLBA in Section 4.1.4 the Euler-Lagrange equations have to be discretized. Therefore, the abbreviations

$$\begin{aligned} \psi_{dmn}' &:= \psi_d'\left(\tilde{\boldsymbol{\alpha}}_{mn}^T J_{mn} \tilde{\boldsymbol{\alpha}}_{mn}\right) \\ \psi_{smn}' &:= \psi_s'\left(\|\nabla \boldsymbol{\alpha}_{mn}\|_2^2\right) \end{aligned}$$

which describe the derivatives of the error functions at location (m, n) are used. Accordingly, the discrete image space is again given by $(m, n) \in \mathbb{N}^2$ with distance h between neighboring points, and N_x and N_y denoting the number of locations in horizontal and vertical direction, respectively. Furthermore, $\mathcal{N}(m, n)$ denotes the set of the four nearest neighbors of pixel (m, n) . The second

term of Equation (4.21) is similar to the non-linear diffusion equation $\partial_t c = \operatorname{div}\left(D\left(|\nabla c|^2\right)\nabla c\right)$ with concentration c and diffusivity function D [Brox, 2005]. According to Brox [2005], the discrete version of this term is thus given by

$$\sum_{\tilde{m},\tilde{n}\in\mathcal{N}(m,n)} \frac{\psi'_{s\tilde{m}\tilde{n}} + \psi'_{smn}}{2} \frac{\alpha_{\tilde{m}\tilde{n},i} - \alpha_{mn,i}}{h^2} \ . \tag{4.23}$$

With the above definitions and approximations the discretized Euler-Lagrange equations at pixel (m,n) can be formulated as

$$\begin{aligned} &\frac{\psi'_{dmn}}{\lambda}\left(\sum_{j=1}^{K} J_{mn,ij}\alpha_{mn,j} + J_{mn,iK+1}\right) \\ &\quad - \sum_{\tilde{m},\tilde{n}\in\mathcal{N}(m,n)} \frac{\psi'_{s\tilde{m}\tilde{n}} + \psi'_{smn}}{2} \frac{\alpha_{\tilde{m}\tilde{n},i} - \alpha_{mn,i}}{h^2} = 0 \ , \qquad i = 1,\dots,K \ . \end{aligned} \tag{4.24}$$

Because of the dependency of ψ'_{dmn} and ψ'_{smn} on $\boldsymbol{\alpha}_{mn}$ the discrete Euler-Lagrange equations are still non-linear. Therefore, an outer fixed point iteration is applied in order to remove this non-linearity. To do so, ψ'_{dmn} and ψ'_{smn} are kept fixed while the resulting linear system of equations is solved for $\boldsymbol{\alpha}_{mn}$ using for instance SOR. The iterative solution of the linear system is given by

$$\begin{aligned} \alpha_{mn,i}^{(q+1)} &= (1-\omega)\,\alpha_{mn,i}^{(q)} \\ &\quad + \omega\, \frac{\sum\limits_{\tilde{m},\tilde{n}\in\mathcal{N}^-(m,n)} \frac{\psi'_{s\tilde{m}\tilde{n}}+\psi'_{smn}}{2}\alpha_{\tilde{m},\tilde{n},i}^{(q+1)} + \sum\limits_{\tilde{m},\tilde{n}\in\mathcal{N}^+(m,n)} \frac{\psi'_{s\tilde{m}\tilde{n}}+\psi'_{smn}}{2}\alpha_{\tilde{m},\tilde{n},i}^{(q)}}{\sum\limits_{\tilde{m},\tilde{n}\in\mathcal{N}(m,n)} \frac{\psi'_{s\tilde{m}\tilde{n}}+\psi'_{smn}}{2} + \psi'_{dmn}\frac{h^2}{\lambda}J_{mn,ii}} \\ &\quad - \omega\, \frac{\psi'_{dmn}\frac{h^2}{\lambda}\left(\sum\limits_{j=1,j<i}^{K} J_{mn,ij}\alpha_{mn,j}^{(q+1)} + \sum\limits_{j=1,j>i}^{K} J_{mn,ij}\alpha_{mn,j}^{(q)} + J_{mn,iK+1}\right)}{\sum\limits_{\tilde{m},\tilde{n}\in\mathcal{N}(m,n)} \frac{\psi'_{s\tilde{m}\tilde{n}}+\psi'_{smn}}{2} + \psi'_{dmn}\frac{h^2}{\lambda}J_{mn,ii}} \end{aligned} \tag{4.25}$$

for $i = 1,\dots,K$, $m = 1,\dots,N_y$ and $n = 1,\dots,N_x$. Then $\boldsymbol{\alpha}_{mn}$ is used to update the values of ψ'_{dmn} and ψ'_{smn} and (4.25) must be solved again. The entire process is repeated until the values of ψ'_{dmn} and ψ'_{smn} are constant and a fixed point is reached. Because the error functions ψ_d and ψ_s are chosen to be convex, the whole problem is convex too and, therefore, has a unique solution [Brox et al., 2004].

Apart from in point-based notation the robust Euler-Lagrange equations can of course also be written in matrix notation. To this end, the procedure is similar to the one in Section 4.1.4.

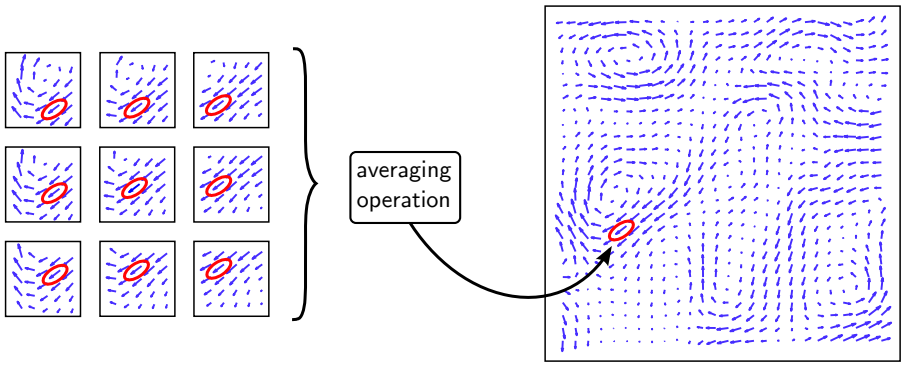

Figure 4.6: Principles of the statistical approach. Each flow vector of the flow field is derived via an average operation from several reconstructed flow patches. The reconstructed patches are thereby derived at slightly different locations.

4.1.6 Statistical approach

All approaches introduced in the previous sections only use the central flow vector of the reconstructed patch as shown in Figure 4.2. The complete information given by the surrounding flow vectors is discarded. However, this information can be used in a statistical manner to improve the results by having not just one flow estimate per pixel but several. The maximum number of estimates per final flow vector is equal to the total number of flow vectors in the reconstructed patches. Yet, also smaller samples consisting of a defined number of vectors around the central vector can be used.

The principles are sketched in Figure 4.6. Several reconstructed flow patches are used to obtain one flow vector. Accordingly, the parameter vector $\boldsymbol{\alpha}$ of each patch is estimated from slightly different neighborhoods of the image sequence applying any variant of the LBA. By using the linear combination (4.1) of parameters α_k and motion models ϕ_k one patch of flow vectors is obtained for each neighborhood as demonstrated in Figure 4.2. Normally, only the central vector of each reconstructed patch is used as flow estimate. Thus, two neighboring flow vectors $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ are derived from two slightly different neighborhoods named ω_1 and ω_2, respectively, whereas ω_2 is translated by one pixel compared to ω_1. However, there are actually two estimates for the flow vector $\boldsymbol{u}_1$, one given by the central vector of the patch reconstructed from ω_1, and one given by the vector next to the center of the patch reconstructed from ω_2. As shown in Figure 4.6, different flow estimates of the same flow vector obtained from the vectors at different positions of the diverse reconstructed patches are used in combination to determine the respective flow vector. To obtain one flow vector out of the sample of estimates, an averaging operation is applied in which one of the following concepts is used. Let $x_1 \dots x_n$ denote the data in the sample.

- The *arithmetic mean* is given by $\bar{x}_{\text{mean}} = \frac{1}{n} \sum_{i=1}^{n} x_i$.
- The *median* is the value that separates the data sample into a lower and an upper half.

For a sorted sample it is given by $\bar{x}_{\text{med}} = x_{\left(\frac{n+1}{2}\right)}$.

- The *weighted arithmetic mean* is given by $\bar{x}_{\text{wmean}} = \frac{\sum_{i=1}^{n} w_i \cdot x_i}{\sum_{i=1}^{n} w_i}$ with weights w_i. For the statistical LBA, the central pixel, which is the sole pixel used in the normal LBA, is weighted equally strong as all other pixels together.

- The *truncated arithmetic mean* is similar to the arithmetic mean where a certain number of the highest and an equal number of the lowest values is cut off. In the case of outliers, the truncated mean is more robust than the normal arithmetic mean. For the statistical LBA, the cut off refers to 20 % of the highest and lowest values, respectively.

- For the purpose of this work, the *weighted and truncated mean* is a combination that weights the central pixel equally strong as the truncated mean of all other pixels.

In addition to these simple statistical averaging methods also more advanced techniques, based on the probability density function of the data sample could be applied. Considering that the sample data does not necessarily follow a normal distribution, more advanced techniques could further improve the method.

4.2 Implementation details

The implementation of optical flow approaches has great influence on the quality of the estimated flow fields. Small changes in the code such as for instance the application of different filter kernels may have a tremendous effect on the results. Different implementations of the same optical flow model may lead to results that strongly differ from each other [Sun et al., 2014]. Therefore, the algorithmic realization of the LBA has to be designed carefully and the influence of some implementation details must be considered.

The approach was implemented in MATLAB, which is a high-level language and an interactive environment for numerical computation, visualization, and programming. Together with the Image Processing Toolbox™ the software is a great tool for doing image processing. It has many useful built-in functions such as SVD and supports vector as well as matrix operations. Program writing and algorithm implementation is simple and fast, a reason why it is comfortable to use for the development of new processes. Compared to languages like C/C++, MATLAB can be quite slow, particularly when many for-loops are used. However, for this project an easy adaptability was more important than computation time and, therefore, the choice of MATLAB is not critical.

4.2.1 Algorithm

A schematic overview of the algorithmic realization of the LBA is shown in Figure 4.7. It consists mainly of three parts described in the following section.

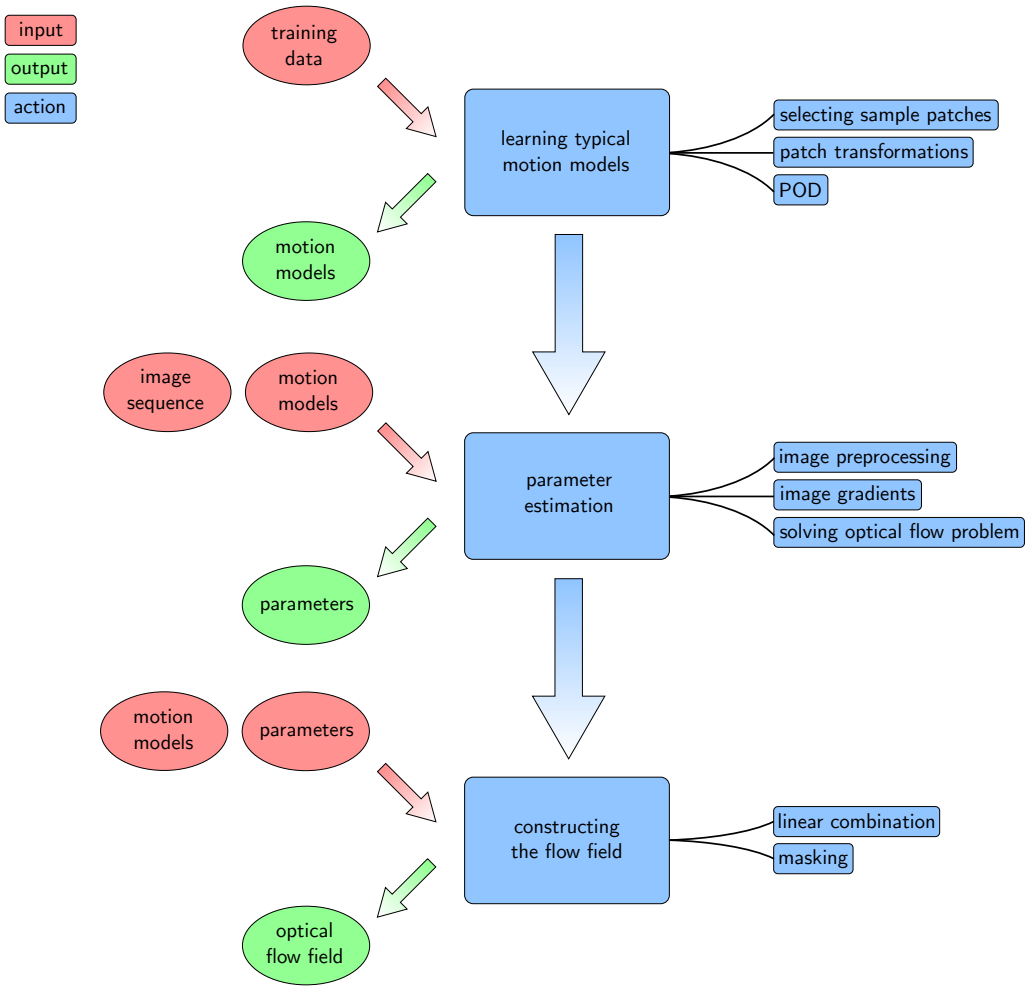

Figure 4.7: Schematic overview of the processing pipeline. The different algorithmic components are displayed in blue. Inputs and outputs are shown in red and green, respectively.

Learning typical motion models

The learning process is implemented as described in Section 4.1.2. From the training vector fields, which serve as input, the sample patches are randomly chosen. These patches are altered in the patch transformation block. The output in form of the typical motion models ϕ_k is determined via POD. Notably, when appropriate motion models are known from a previous learning process, the entire block can be omitted.

Parameter estimation

This is the main part of the algorithm. The parameters that define the displacement field are estimated from the image sequence in dependence of the learned motion models. In some cases, as for instance noisy images, preprocessing of the images yields improved optical flow fields. Therefore, a smoothing operation in form of a convolution with a Gaussian kernel K_σ of standard

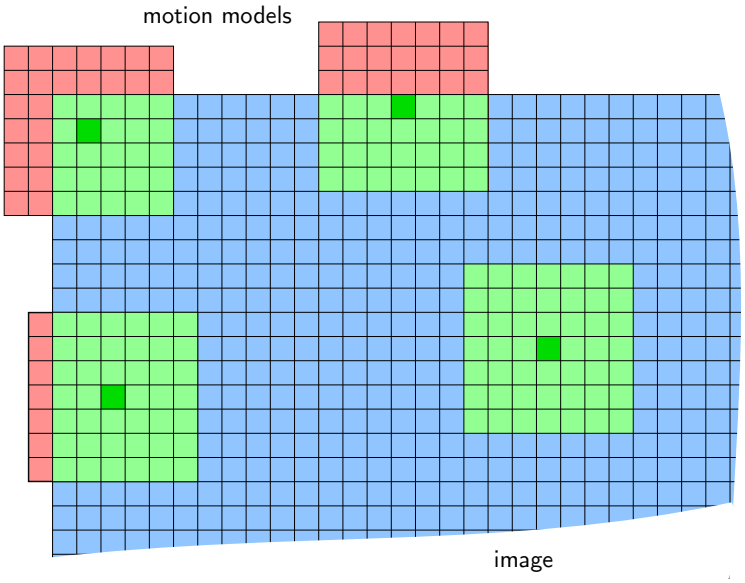

Figure 4.8: Illustration of the boundary pixel problem. At some pixel locations at the boundary of the image it is not possible to use the entire motion models because they extend beyond the boundaries of the image. At these locations the motion models are shortened by cropping the red area. At locations where the motion models fit completely into the image, they can be used as a whole.

deviation σ is conducted (cf. Section 2.3.4). By the image smoothing the high frequencies that are more error-prone than low frequencies are removed [Scharr, 2000]. With regard to particle images, it also diffuses distinct particles isotropically and connects corresponding particles from successive images, which is a requirement to obtain correct gradient images [Liu and Shen, 2008]. Gaussian smoothing is particularly of advantage for larger displacements. However, smoothing must be applied carefully, because important image structures could be smoothed out and destroyed. The size of the Gaussian kernel is given by $\rho = 2 \cdot \lceil 3\sigma \rceil + 1$ with the ceil operator $\lceil x \rceil = \min \{k \in \mathbb{Z} \,|\, k \geq x\}$. Sun et al. [2014] compared a variety of preprocessing operations and showed that the relatively simple Gaussian smoothing performs well compared to other more advanced preprocessors.

In order to determine the gradient images, different filter operations are performed. Scharr [2000] demonstrated that the choice of the filter kernel may have a large effect on the resulting flow fields. All image derivatives are usually computed with the gradient filters optimized for optical flow by Scharr [2007]. In the case of equidistant time steps between the single frames of the image sequence, filters of size $5 \times 5 \times 5$ are used. The filter kernel consists of the derivator stencil $[0.0836, 0.3327, 0, -0.3327, -0.0836]$ in direction of the derivation and the smoother stencil $[0.0233, 0.2415, 0.4704, 0.2415, 0.0233]$ in the two remaining directions. In the case of image pairs a $5 \times 5 \times 2$ filter with simple two point stencil $[1, -1]$ in time direction is applied. At the boundaries of the images the values of the boundary pixels are mirrored. This is necessary to obtain images of full size, because the filter kernel extends beyond the boundary of the images (cf. Section 2.3.4). However, it is important to bare in mind, that the boundary values of the gradient images, which could be determined in this way, may not be correct.

The optical flow problem itself is solved by one of the approaches introduced in Section 4.1. Options are to imply the local, the variational or the robust variational approaches as well as

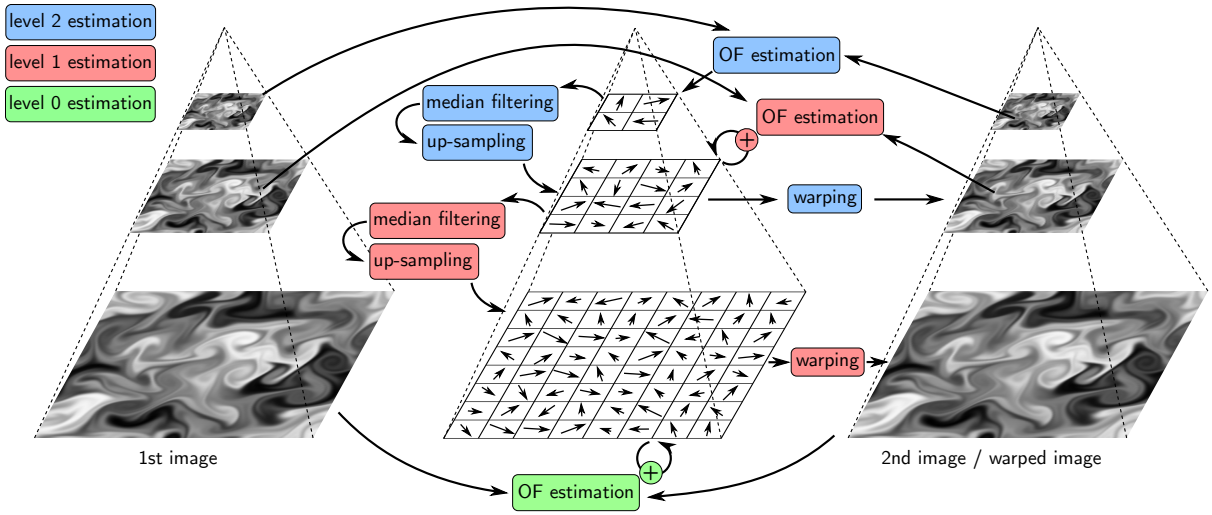

Figure 4.9: Schematic drawing of the hierarchical optical flow approach with three levels. The algorithm starts at the top pyramid level ($q = 2$) and continues towards finer levels until the bottom level ($q = 0$) is reached. The images are taken from the scalar images of the 2D turbulent sequence provided by Carlier [2005b].

their statistical extensions. However, the output of the function is not the optical flow field itself but a 3D array that defines a parameter vector $\boldsymbol{\alpha}$ at each pixel location. Different parts of the motion models are utilized depending on the location of the current pixel within the image. This is necessary in order to obtain flow fields of the same size as the images. Otherwise it would not be possible to determine optical flow vectors at boundary locations. The problem is illustrated in Figure 4.8.

Constructing the flow field

The optical flow is given by the linear combination (4.1) of the estimated parameters and the learned motion models. In the case of the normal LBA, the calculation of the optical flow field is straight forward as described in Section 4.1.1. If the statistical LBA is used, the calculation is only slightly more complicated (cf. Section 4.1.6).

The masking part is optional. In the case of stationary, visible flow boundaries in the images (e.g., sequence 'a6' described in Section 4.3.1), a binary masking image is used to set the flow vectors within the boundary to zero. Therefore, the optical flow field is pixel wise multiplied by the masking image, which consists of 'zeros' at locations on the boundary and the object and 'ones' at all other locations.

4.2.2 Hierarchical approach

In the case of relatively large displacements a hierarchical multi-scale approach is applied (cf. Section 3.4.6). This is necessary because the linearized BCCE (3.6) is only valid for small displacements. The principles of the hierarchical pyramid approach are sketched in Figure 4.9.

The basic steps are:

1. The Gauss pyramids of the two successive images I_1 and I_2 are calculated by

$$I_p^{(q)} = (K_\sigma * I_p^{(q-1)})_{\downarrow r}$$

   with Gaussian kernel K_σ of standard deviation σ. The subscript $p \in \{1, 2\}$ denotes the image, q denotes the pyramid level, and $\downarrow r$ denotes the down sampling rate (cf. Section 3.4.6). The value of the used standard deviation depends on the down sampling rate, it is: $\sigma = \frac{1}{\downarrow r} - 1$.

2. The optical flow field on the topmost pyramid level (image with lowest resolution) is calculated.

3. Median filtering of the flow field.

4. Up-sampling of the flow field to fit the dimensions of the next finer pyramid level.

5. Backward warping of I_2 towards I_1 using the up-sampled optical flow field.

6. The residual motion field between the warped image I_w and I_1 is estimated. To combine the new flow increment with the actual flow field derived from previous steps, the flow fields are added up.

7. Steps 3 - 6 are repeated until the bottom of the pyramid (image with highest resolution) is reached. Finally the resulting optical flow field is given by the sum of the motion increments from all pyramid levels.

The estimation of the optical flow field in steps 2 and 6 is done as described by the parts *parameter estimation* and *constructing the flow field* shown in Figure 4.7 and described in Section 4.2.1. According to Sun et al. [2014], the median filtering performed during step 3 of the intermediate flow field on every pyramid level is a key implementation detail, which strongly improves the quality of the resulting optical flow field. In order to perform the image warping in step 5, interpolation is necessary, in which the values of the flow vectors must be evaluated between pixels due to their sub-pixel length. Usually, a bicubic interpolation is used.

4.3 Testing the approach

In order to investigate the properties and the performance of the LBA different tests were conducted. Therefore, the optical flow of several synthetically generated particle image sequences, introduced in Section 4.3.1, was estimated. The influence of the model size on the resulting optical flow field is observed in Section 4.3.3. The effect of the number and the composite of the used motion models is analyzed in Section 4.3.4. In Section 4.3.5, the influence of the training

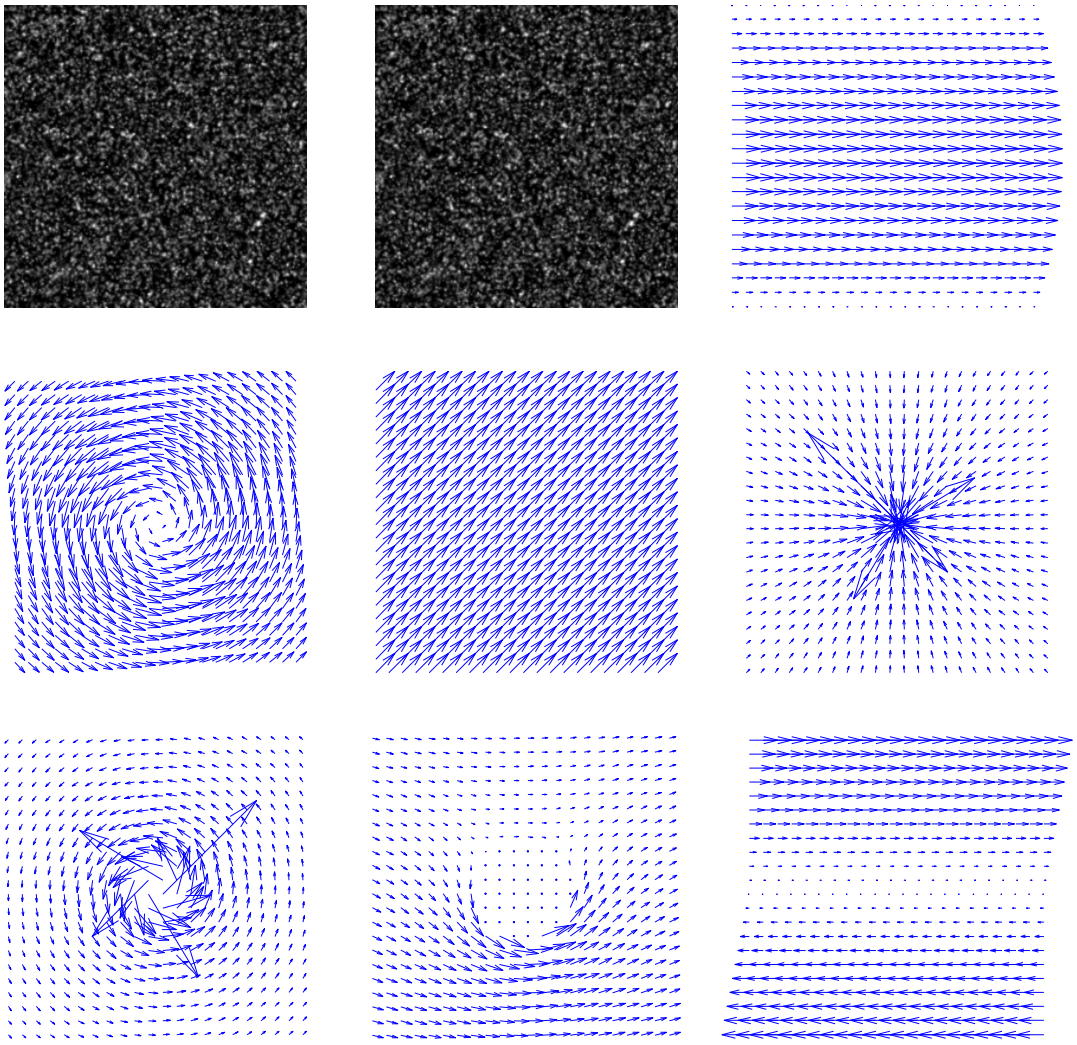

Figure 4.10: The analytic set. From *top* to *bottom* and *left* to *right*, the first and the second image of sequence 'a1' and the ground truth of sequence 'a1', 'a2', 'a3', 'a4', 'a5', 'a6', and 'a8' are shown, respectively.

data used to learn the motion models is explored. The statistical versions of the LBA as well as the rvLBA are analyzed in Section 4.3.6 and Section 4.3.7, respectively. Finally, the computation times of the approaches are compared in Section 4.3.8. Within these tests, no presmoothing of the images was applied. With regard to the many acronyms used in this section the reader is referred to the list of abbreviations on page 121.

4.3.1 The test sequences

The *analytic* set provided by Carlier [2005b] was used as test sequences. The set is freely available and can be downloaded from the FLUID project page (http://fluid.irisa.fr). It consists of seven synthetically generated particle image sequences namely 'a1', 'a2', 'a3', 'a4', 'a5', 'a6', and 'a8'.

Table 4.1: Characteristics of the image sequences of the analytic set such as the dynamic range, the mean velocity, and the standard deviation (std).

name	flow typ	dynamic range $\left[\frac{\text{px}}{\text{frame}}\right]$	mean $\left[\frac{\text{px}}{\text{frame}}\right]$	std $\left[\frac{\text{px}}{\text{frame}}\right]$
'a1'	Poiseuille	0.02 - 2	1.33	0.59
'a2'	Lamb-Oseen vortex	0.01 - 0.55	0.45	0.08
'a3'	uniform	$\sqrt{2}$ - $\sqrt{2}$	$\sqrt{2}$	0
'a4'	sink	0.06 - 14.75	0.14	0.21
'a5'	vortex	0.06 - 14.75	0.14	0.21
'a6'	cylinder with rotation	0 - 3.40	1.02	0.60
'a8'	gradient	0 - 0.50	0.25	0.14

A sequence called 'a7' does not exist in the data set. Each sequence contains 41 images of size $256 \times 256\,\text{px}^2$ in 8-bit TIFF format. In order to generate the initial images of each sequence, the same parameters such as particle size and particle concentration were used. The mean particle diameter is 1.5 px and the *particle number density* (PND) is approximately 0.25 ppp (particles per pixel). The image sequences are derived from computed flow fields that are also provided together with the image sequences. They represent the correct solutions, (*ground truth*) of the flow problems. In Figure 4.10 two subsequent frames of the first sequence are shown together with the correct flow fields of all seven sequences.

Some properties of the flow fields are listed in Table 4.1. The Poiseuille ('a1'), the Lamb-Oseen ('a2'), and the gradient ('a8') flow are examples of viscous flows. Their velocity field directly results from the solution of the Navier-Stokes equations. The Poiseuille flow describes the flow between two parallel plates. The Lamb-Oseen vortex contains circular streamlines and radial decreasing vorticity. The uniform ('a3'), the sink ('a4'), and the vortex flow ('a5'), as well as the flow around a cylinder with rotation ('a6') are potential flows and, therefore, the velocity field is given by the gradient of the scalar potential field. One characteristic of potential flows is that they are incompressible ($\nabla \cdot \boldsymbol{u}$) and irrotational ($\nabla \times \boldsymbol{u}$). The uniform flow contains flow from bottom left to top right. The sink and the vortex is arranged at the corresponding image center. According to Carlier [2005b], the flow around a cylinder with rotation is a superposition of a uniform doublet and a vortex flow.

The particle displacement is mostly below $2\,\frac{\text{px}}{\text{frame}}$. Only at the center of the sink ('a4') and at the center of the vortex flow ('a5'), as well as near the cylinder wall of the cylinder flow ('a6') the displacement extends this value in small local areas. Due to a linearization of the data term, relatively small displacements are a requirement for most optical flow approaches. Otherwise, multi-scale techniques must be applied in order to cope with large displacements (cf. Section 3.4.6). Since there are only a few locations where the displacement is larger than $2\,\frac{\text{px}}{\text{frame}}$ the simple LBA without multi-scale methods was used for the test sequences.

The analytic set is considered to be a valuable test case for the LBA because it contains many interesting, yet simple flow aspects such as gradients, curls, and sinks. Furthermore, the

different flow fields of the set can be used to learn different motion models, and their suitability to estimate the flow field of each sequence can be studied. A further advantage of the analytic set is that the correct flow field is known and can be compared to the estimated flow field. Therefore, the performance of the approach can easily be quantified using for instance the AAE introduced in Section 3.5.

4.3.2 The learned motion models

Different sets of motion models were obtained from the correct flow fields of the analytic sequences. Each correct flow field is used as training data, from which the motion models were learned, as described in Section 4.1.2. Additionally, a *mixed* set of motion models was learned from training data that consisted of all seven correct flow fields. In Figure 4.11, the first $k = 9$ learned motion models of size $\omega = 11 \times 11 \times 1$ are shown. Apart from the pictured $11 \times 11 \times 1$ models, also other sets of motion models with different spatio-temporal sizes were learned. The spatial dimension ranged from $\rho = 5$ to $\rho = 21$ and the temporal dimension from $t = 1$ to $t = 7$.

At first glance, it appears that the first two motion models of all sequences shown in Figure 4.11 contain almost translational motion in vertical and horizontal direction. Yet, especially the models of 'a4' and 'a5' reveal some local variations of the flow vectors within the models. The motion models two to nine from each sequence are considerably different from each other. They share in common that local structures with smaller scales are contained in the higher models. In some sequences, special models reminiscent of affine motion models are contained. For instance the single rotation mode is present in the set of 'a2', 'a5', 'a6', and the 'mixed' sequence. Other affine like models contained in some sequences are divergence, stretching, and shear. However, the learned motion models provide some advantages compared to affine models. They may contain, on the one hand, models that are suited to represent complex flow structures, which cannot be represented by affine models, and on the other hand, only contain the fundamental models tailored to the specific problem. In Figure 4.12 spatio-temporal motion models of size $11 \times 11 \times 7$ of sequence 'a6' are depicted. However, these spatio-temporal models contain almost no temporal variation, since the flow field is stationary.

For motion models embodied by later POD modes, the energy or information content becomes smaller. This can be seen in Figure 4.13 where the RIC (cf. Section 2.2.4) is shown for the different sequences. As shown in Equation (2.12) the RIC defines the fraction of the total energy that can be represented by the first K motion models. Figure 4.13 shows that the convergence rates are different for the different sequences. For some sequences such as 'a1', 'a2', 'a3' and 'a8', the first two to five motion models are sufficient to represent virtually the whole energy of the corresponding sequence. In the case of sequence 'a3' the first two models represent literally 100 % of the complete information, which is anticipated, because the sequence only contains uniform motion with no variations. Thus, the models three to nine of 'a3' displayed in Figure 4.11 are somewhat meaningless. The sequences 'a4', 'a5', 'a6', and 'mixed' require even more models than those nine shown in Figure 4.11 in order to represent the entire energy of the

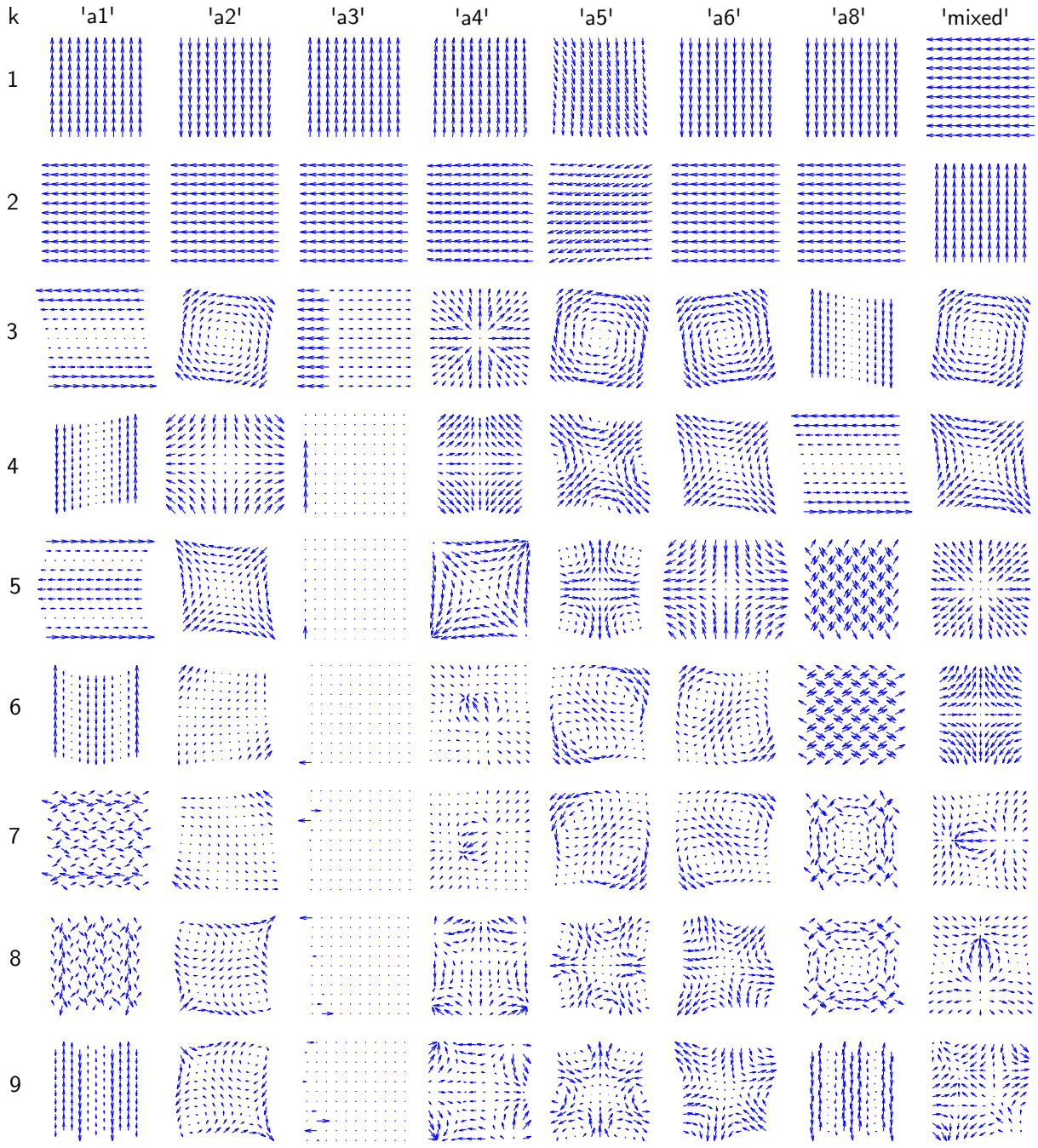

Figure 4.11: The first nine motion models of size $\omega = 11 \times 11 \times 1$ learned from the correct flow fields of the different sequences of the analytic set. Column 8 contains motion models that were learned from a mixed set of all seven sequences. The length of the vectors is individually scaled for each POD mode.

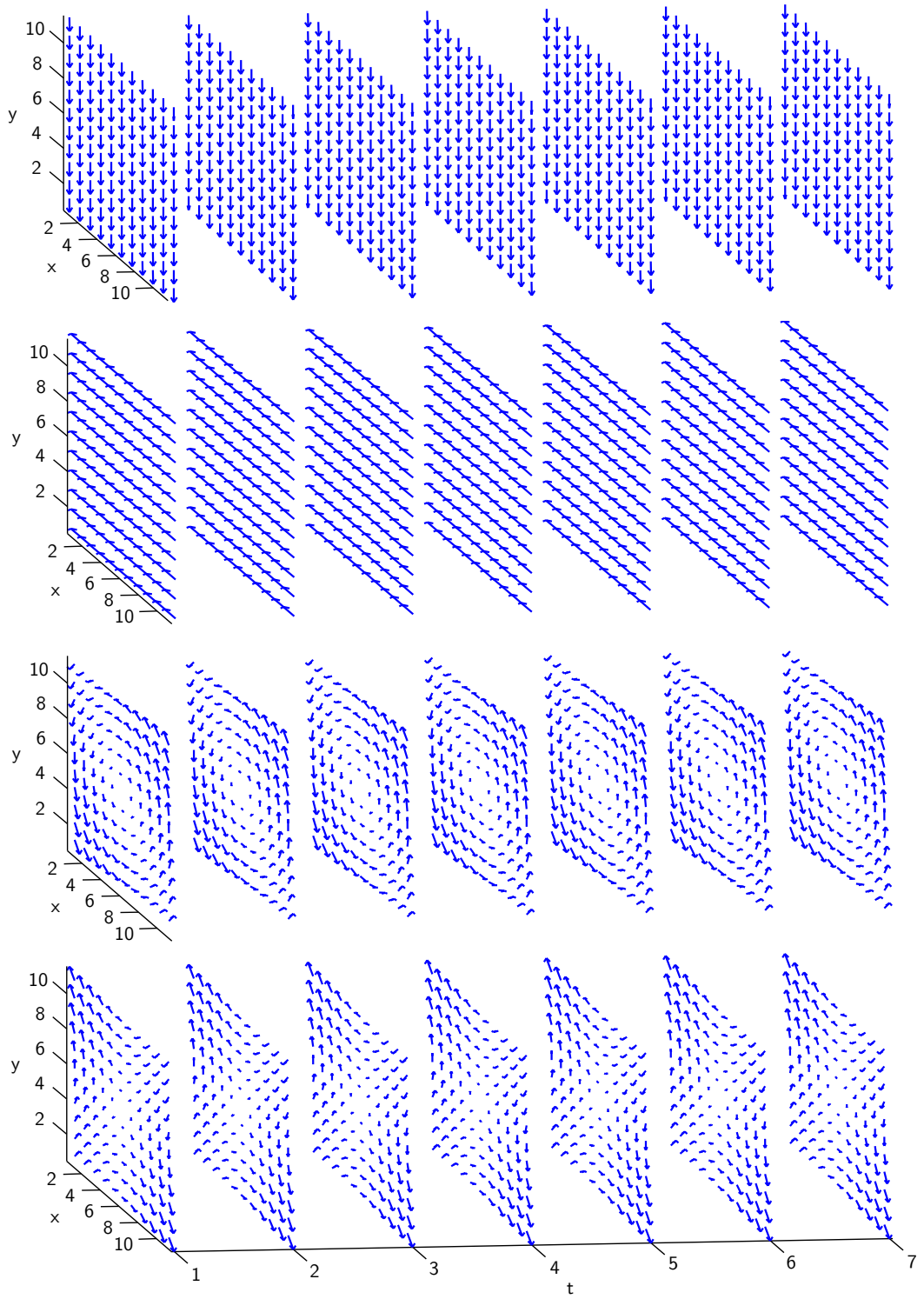

Figure 4.12: The first four motion models of size $\omega = 11 \times 11 \times 7$ learned from the correct flow field of sequence 'a6'. Because the flow field is stationary there are almost no variations between different time steps.

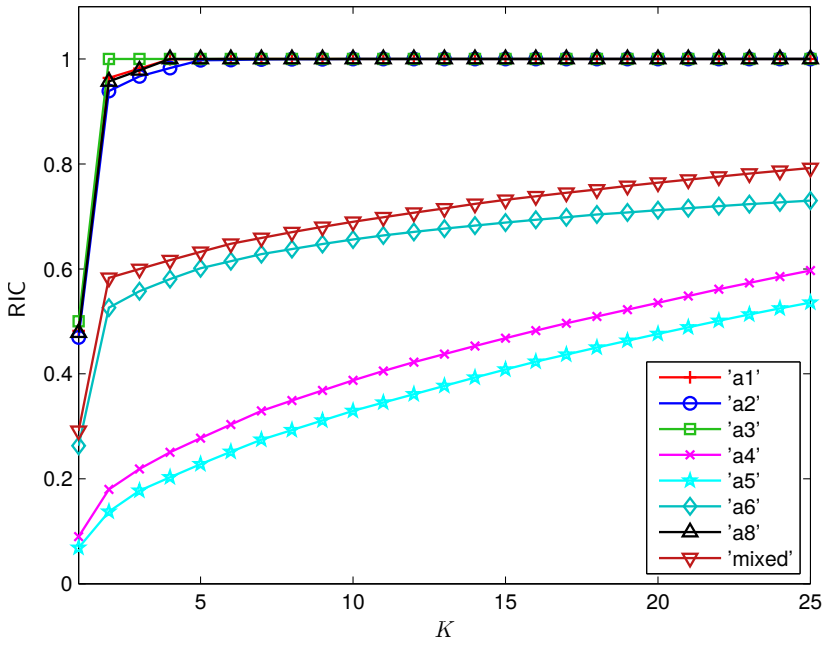

Figure 4.13: RIC of the different sequences of the analytic set shown for the first $K = 25$ motion models of size $11 \times 11 \times 1$.

corresponding sequence. Especially for 'a4' and 'a5' this may be due to the very center of the sequences where many variations, strong gradients, and relatively large motions are present. It seems that most relevant motions are covered by the first few models as evident from the motion models in Figure 4.11.

4.3.3 Influence of the model size

In order to investigate the influence of the size of the learned motion models on the resulting optical flow field, the flow fields of the seven image sequences of the analytic set were estimated with varying model sizes with the LBA. For each sequence, the motion models obtained by using the own correct flow field as training data, were used. The first nine motion models of size $\omega = 11 \times 11 \times 1$ are displayed in Figure 4.11. These *own* motion models are perfectly suited for the particular sequence, since they innately contain all relevant flow structures.

To quantify the performance of the LBA, the AAE defined in Section 3.5.1 was calculated. The results are given in Table 4.2 and Table 4.3 for the lLBA and the vLBA, respectively. The AAE is quoted together with its standard deviation. To obtain these results the number of used motion models was kept fixed to $K = 6$. Yet, in many cases this is not the optimal choice for K as shown in Section 4.3.4, where the influence of the number of used motion models is investigated. The minimal reached AAE of each sequence is highlighted in blue. Additionally, the dependency of the AAE on the model size is shown for the lLBA and the vLBA in Figure 4.14 and Figure 4.15, respectively. Because the plots of 'a2' and 'a8' look very similar to the one of 'a1' and the plot of 'a4' looks similar to the one of 'a5', they are not shown.

As can be seen from both tables and both figures, the AAE essentially decreases with increasing size of the motion models. Increasing the model size leads to a higher number of constraint

Table 4.2: Dependency of the ILBA on the model size for the analytic set. The spatial size is denoted by ρ and the temporal size by t. In all cases the number of used motion models was $K = 6$. The minimal AAE of each sequence is highlighted in blue.

size		AAE [°]						
ρ	t	'a1'	'a2'	'a3'	'a4'	'a5'	'a6'	'a8'
5	1	1.34 ± 1.41	0.67 ± 0.48	0.42 ± 1.38	0.69 ± 2.07	0.65 ± 2.09	1.69 ± 5.20	0.49 ± 0.45
5	3	0.75 ± 0.60	0.41 ± 0.32	0.09 ± 0.63	0.46 ± 1.48	0.92 ± 1.38	0.80 ± 2.15	0.29 ± 0.32
5	7	0.48 ± 0.43	0.24 ± 0.22	0.06 ± 0.44	0.64 ± 1.98	0.79 ± 1.66	0.52 ± 1.18	0.16 ± 0.22
11	1	0.39 ± 0.34	0.29 ± 0.24	0.12 ± 0.57	0.91 ± 1.85	0.58 ± 1.69	0.71 ± 2.55	0.23 ± 0.28
11	3	0.28 ± 0.24	0.18 ± 0.21	0.09 ± 0.43	0.34 ± 1.31	0.59 ± 1.09	0.45 ± 1.10	0.14 ± 0.21
11	7	0.22 ± 0.23	0.12 ± 0.18	0.09 ± 0.38	0.36 ± 1.25	0.20 ± 1.42	0.31 ± 0.78	0.08 ± 0.18
15	1	0.27 ± 0.26	0.22 ± 0.19	0.12 ± 0.45	0.42 ± 1.42	0.44 ± 1.56	0.50 ± 1.52	0.17 ± 0.22
15	3	0.20 ± 0.20	0.14 ± 0.17	0.10 ± 0.37	0.44 ± 1.26	0.36 ± 1.16	0.38 ± 0.86	0.11 ± 0.18
15	7	0.17 ± 0.19	0.10 ± 0.15	0.10 ± 0.33	0.44 ± 1.16	0.23 ± 1.41	0.32 ± 0.69	0.07 ± 0.15
21	1	0.21 ± 0.21	0.18 ± 0.15	0.12 ± 0.36	0.42 ± 1.62	0.42 ± 1.55	0.44 ± 0.91	0.13 ± 0.19
21	3	0.18 ± 0.18	0.13 ± 0.13	0.11 ± 0.31	0.41 ± 1.34	0.35 ± 1.50	0.39 ± 0.74	0.08 ± 0.16
21	7	0.15 ± 0.16	0.10 ± 0.12	0.11 ± 0.28	0.44 ± 1.30	0.54 ± 1.29	0.34 ± 0.65	0.05 ± 0.13

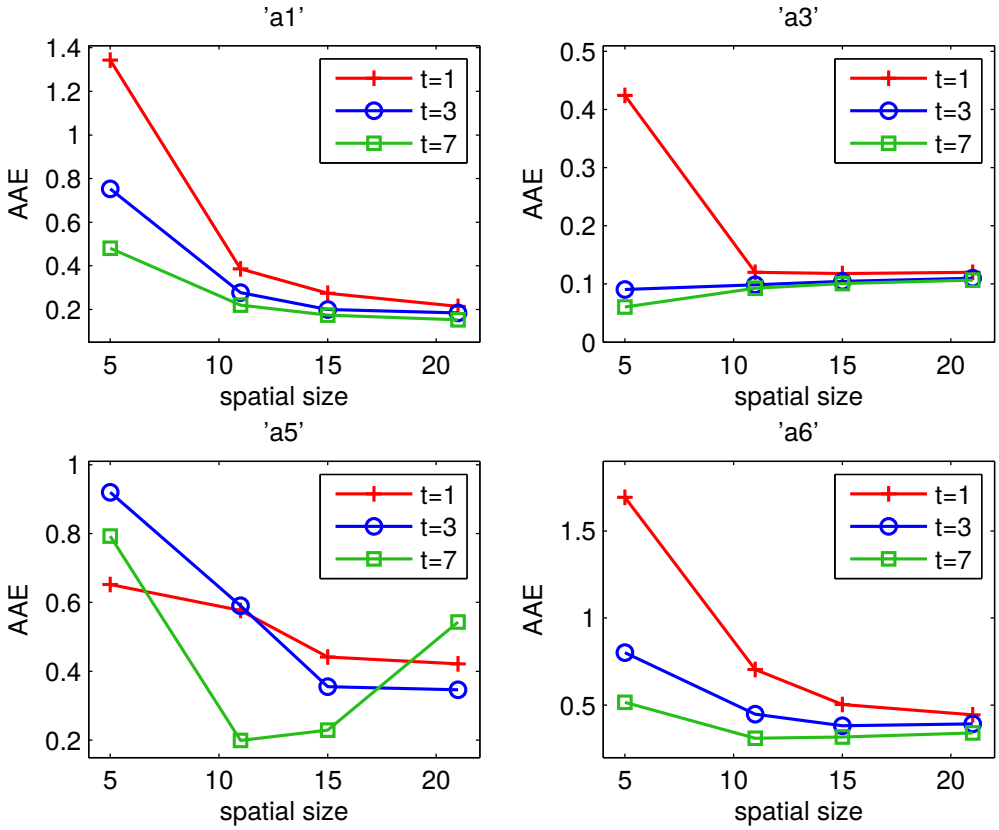

Figure 4.14: Dependency of the AAE of the ILBA on the model size of the sequences 'a1','a3','a5', and 'a6' (from *top left* to *bottom right*). The parameter t denotes the temporal dimension of the motion models. The number of used motion models was $K = 6$.

Table 4.3: Dependency of the vLBA on the model size for the analytic set. The spatial size is denoted by ρ and the temporal size by t. In all cases the number of used motion models was $K = 6$. The minimal AAE of each sequence is highlighted in blue.

size		AAE [°]						
ρ	t	'a1'	'a2'	'a3'	'a4'	'a5'	'a6'	'a8'
5	1	0.27 ± 0.26	0.28 ± 0.15	0.11 ± 0.40	0.35 ± 2.00	0.38 ± 2.42	0.45 ± 1.24	0.14 ± 0.14
5	3	0.24 ± 0.25	0.15 ± 0.14	0.08 ± 0.53	0.27 ± 1.97	0.57 ± 1.49	0.37 ± 1.12	0.10 ± 0.13
5	7	0.21 ± 0.24	0.12 ± 0.12	0.06 ± 0.39	0.50 ± 1.55	0.48 ± 1.56	0.31 ± 0.81	0.06 ± 0.11
11	1	0.23 ± 0.24	0.19 ± 0.14	0.11 ± 0.38	0.55 ± 1.69	0.51 ± 1.78	0.42 ± 1.16	0.13 ± 0.14
11	3	0.20 ± 0.19	0.13 ± 0.13	0.10 ± 0.31	0.27 ± 1.40	0.51 ± 1.25	0.36 ± 0.89	0.09 ± 0.12
11	7	0.17 ± 0.18	0.10 ± 0.14	0.09 ± 0.37	0.28 ± 1.31	0.18 ± 1.53	0.27 ± 0.76	0.06 ± 0.12
15	1	0.22 ± 0.23	0.18 ± 0.13	0.11 ± 0.34	0.33 ± 1.91	0.41 ± 1.76	0.41 ± 1.11	0.12 ± 0.14
15	3	0.17 ± 0.16	0.13 ± 0.12	0.10 ± 0.29	0.41 ± 1.38	0.34 ± 1.44	0.35 ± 0.82	0.08 ± 0.12
15	7	0.15 ± 0.16	0.09 ± 0.13	0.10 ± 0.27	0.42 ± 1.33	0.22 ± 1.53	0.31 ± 0.71	0.06 ± 0.11
21	1	0.18 ± 0.17	0.17 ± 0.12	0.12 ± 0.28	0.40 ± 1.99	0.41 ± 1.75	0.41 ± 0.83	0.11 ± 0.13
21	3	0.15 ± 0.15	0.12 ± 0.12	0.11 ± 0.24	0.38 ± 1.78	0.34 ± 1.59	0.38 ± 0.77	0.08 ± 0.13
21	7	0.13 ± 0.14	0.10 ± 0.11	0.11 ± 0.23	0.42 ± 1.43	0.54 ± 1.47	0.34 ± 0.67	0.05 ± 0.12

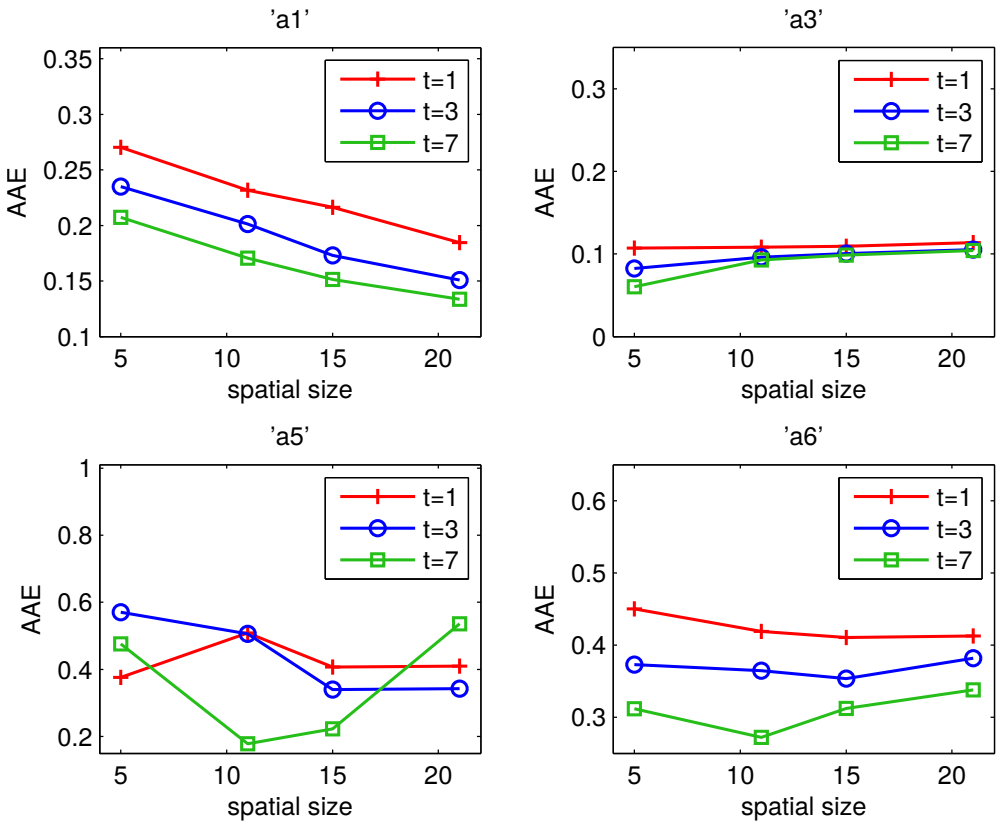

Figure 4.15: Dependency of the AAE of the vLBA on the model size of the sequences 'a1','a3','a5', and 'a6' (from *top left* to *bottom right*). The parameter t denotes the temporal dimension of the motion models. The number of used motion models was $K = 6$.

equations. Because the number of unknown coefficients is equal for different models sizes this leads to higher accuracy. However, in some cases as for instance the sequences 'a5' and 'a6', the AAE is minimal for the size $\omega = 11 \times 11 \times 7$ and increases again for larger model sizes. The minimal AAE of sequence 'a4' is reached for $\omega = 11 \times 11 \times 3$. It is the sole sequence for which the optimal temporal dimension differs from $t = 7$. As the ideal model size also depends on the number of used motion models, this influence is investigated in Section 4.3.4. In this section also the combination of model size and number of used models that yield the lowest error values are given. The sequence 'a3' is somewhat special, because the optimal model size is $\omega = 5 \times 5 \times 7$. This may be due to the fact, that the displacement field shows absolutely no variations and, therefore, an increased model rather introduces some errors than new information. Strikingly, the ideal model sizes are the same for the lLBA and the vLBA.

In Figure 4.16 the AE of the Lamb-Oseen vortex sequence ('a2') is shown for various model sizes for the vLBA. With increasing model size the values of the AE decrease. For spatial size $\rho = 5$ and temporal size $t = 1$, there are some areas of relatively strong errors around the vortex center. These error values become smaller with increasing spatial and temporal dimensions. In accordance with Table 4.3 the error values are lowest for $\rho = 15$ and $t = 7$ and increase again for $\rho = 21$. However, if the number of used motion models is optimized for each spatial and temporal model size, the evolution of the error values is slightly different. The AE values of sequence 'a2' obtained with the vLBA with optimized K for different model sizes is shown in Figure 4.17. A comparison of Figure 4.16 with Figure 4.17 clearly demonstrates that the effect of the number of used motion models on the resulting flow field is not negligible. Therefore, the influence of the number of used motion models on the resulting flow field is investigated in the next section.

4.3.4 Influence of the number of used motion models

To examine the influence of the number of POD modes (K) included in the flow model, Figure 4.18 shows the dependency of the AAE on K for all sequences of the analytic set. The figure contains a plot for the lLBA on the left side and a plot for the vLBA on the right side. The trends of both methods are very similar. Essentially, the estimated optical flow field becomes more accurate with increasing K, as indicated by the decreasing values of the AAE. The effect on the AAE is quite strong for the first few motion models. This implies that these motion models contain the most important flow structures and, therefore, must be included in the set of motion models for an accurate estimation of the optical flow. This is not surprising, because by using POD the motion models were constructed such that the first models contain the most information (cf. Section 2.2). For a particular K, which depends on each sequence, a minimum value of the AAE is reached. A further increase of K leads to a plateau of the AAE or to slightly increased values again. Therefore, it is crucial to choose the number K of POD modes, included in the set of motion models, to be sufficiently large. However, choosing K too large may lead to slightly decreased results and unnecessarily long calculation times as shown in Section 4.3.8.

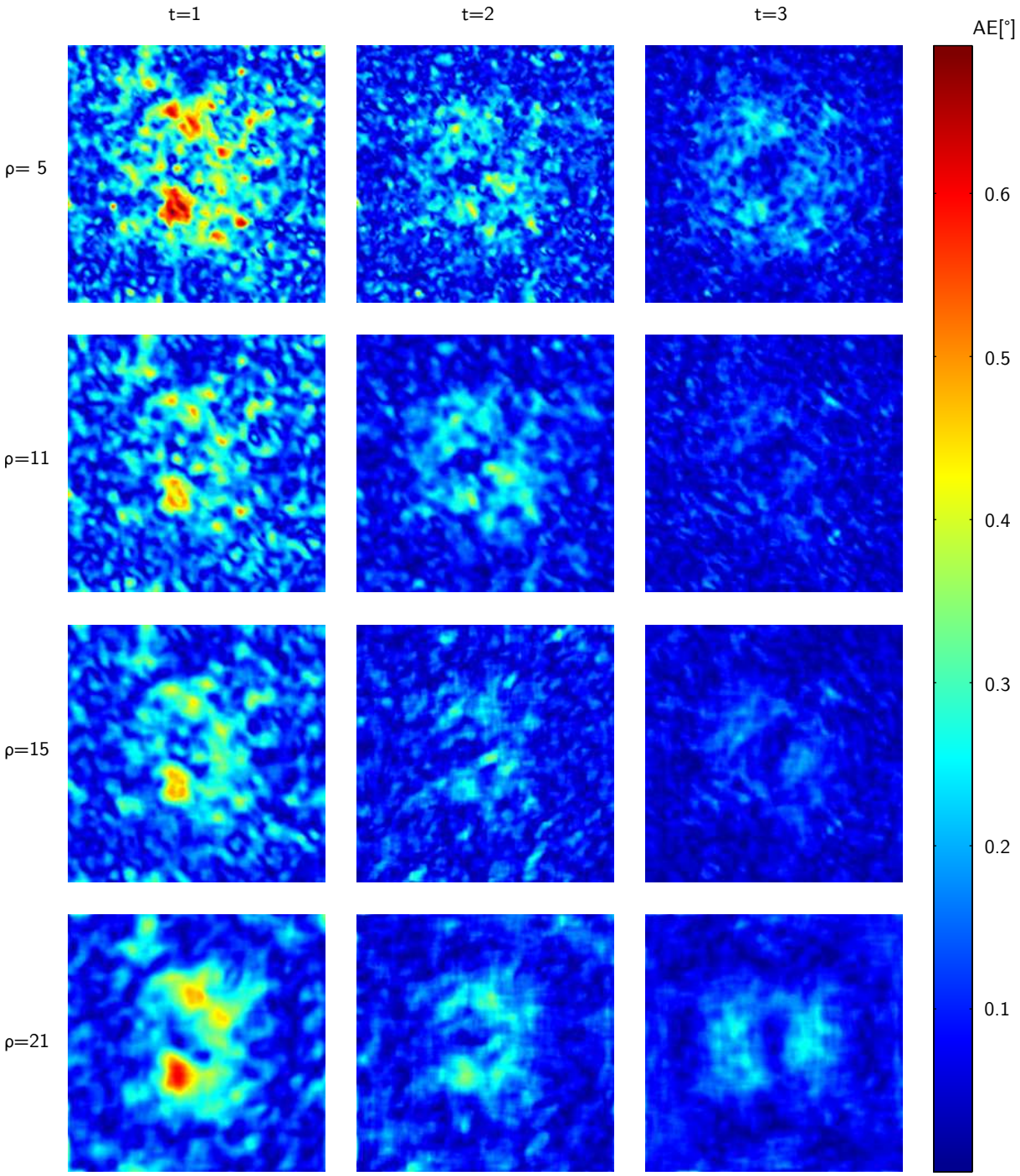

Figure 4.16: AE of sequence 'a2' for different spatial (ρ) and temporal (t) dimensions of the motion models. The optical flow field was estimated with the vLBA. A fixed number of motion models ($K = 6$) was used for each model size. The corresponding AAE values are given in Table 4.3.

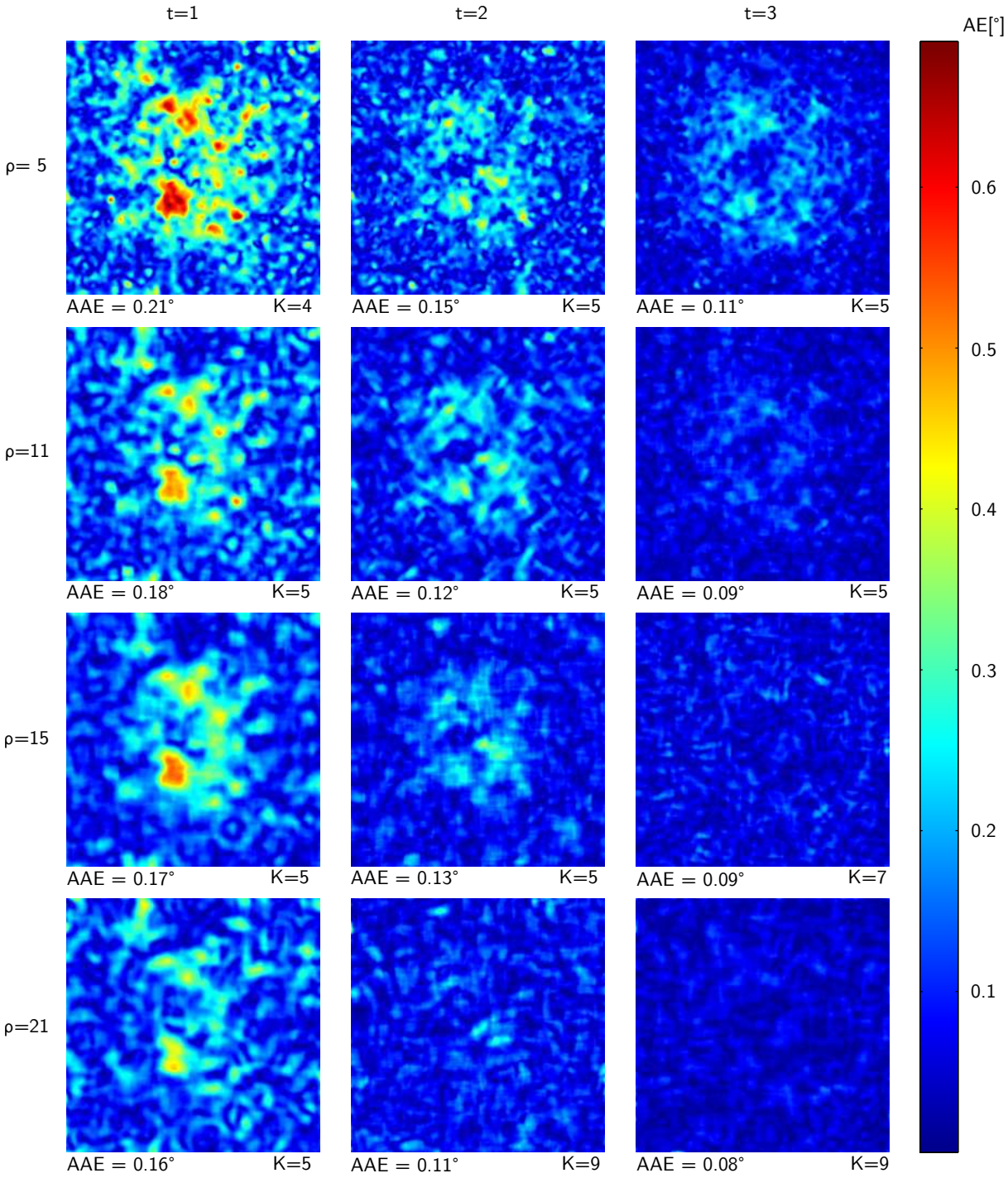

Figure 4.17: AE of sequence 'a2' for different spatial (ρ) and temporal (t) dimensions of the motion models. The optical flow field was estimated with the vLBA. The number of used motion models was optimized for each spatial and temporal model size. Each optimal value of the AAE is shown together with the used number of motion models K below the images.

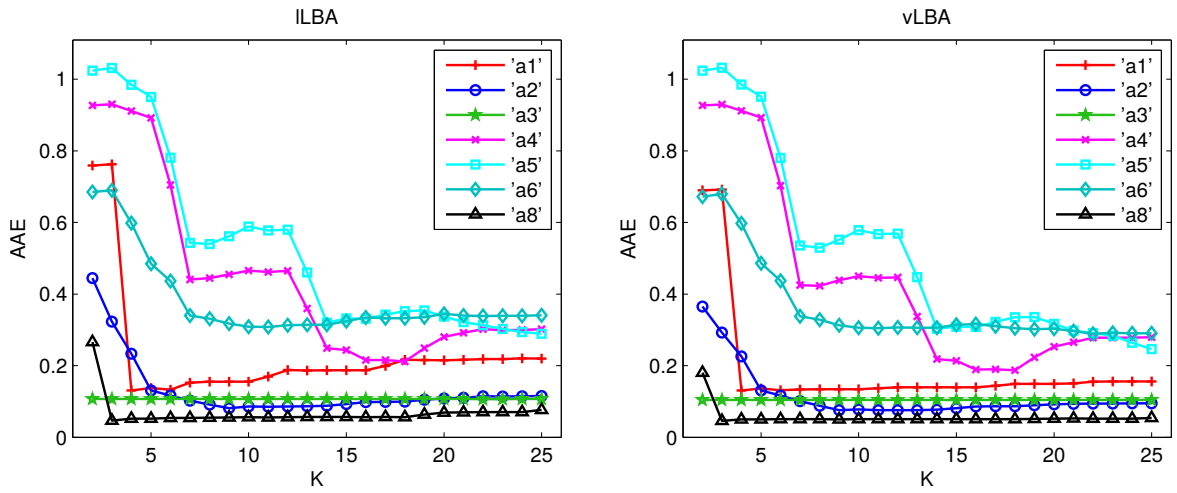

Figure 4.18: Influence of the number of motion models K on the AAE for the lLBA (*left*) and the vLBA (*right*). The size of the used motion models was $\omega = 21 \times 21 \times 7$.

The sequence 'a6' and especially the sequences 'a4' and 'a5' require a larger number of motion models compared to other sequences to reach a minimum of the AAE. The same prediction can be obtained from the plot of the RIC shown in Figure 4.13, which indicates that these three sequences require many more POD modes than the other sequences in order to represent the complete information.

The effect of omitting single motion models on the resulting AAE is shown in Table 4.4. In the case of important POD modes not being included in the set of motion models the AAE is up to several hundred times higher than the AAE obtained using the complete set. In this case the total number of POD modes in the set of motion models was reduced to the first nine modes. The motion models shown in Figure 4.11 are similar to the models used in Table 4.4. Yet, they are not equal, because they are of different dimensions. The models from Table 4.4 are of size $\omega = 15 \times 15 \times 7$ and the models shown in Figure 4.11 are of size $\omega = 11 \times 11 \times 1$. The main difference is the inclusion of temporal information in the models used in Table 4.4. Nevertheless, the $11 \times 11 \times 1$ motion models of Figure 4.11 give a comprehensive overview of the general appearance of the motion models.

The largest effect on the AAE has as expected the omission of one of the first two POD modes, which describe translation like motion in two orthogonal directions. Without these important POD modes an estimation of the flow field is virtually not feasible. For the sequences 'a1' and 'a8', which contain no movement in vertical direction, omitting the first mode, which is responsible for the gross movement in vertical direction, has no negative effect on the result, in fact the AAE is even slightly lower. However, there are also other important modes in most sets of motion models. For instance the third POD mode of the 'a2' model contains rotational motion. Excluding this mode leads to an AAE, which is twice as high as the AAE obtained with all nine motion models. The fourth and the fifth mode of this sequence also have a comparable

Table 4.4: Results in form of the AAE of the lLBA omitting individual motion models. The first column indicates which models were omitted from the set of nine motion models. For all sequences a model size of $\omega = 15 \times 15 \times 7$ was chosen. The results for the vLBA are similar.

omit	AAE [°]						
k	'a1'	'a2'	'a3'	'a4'	'a5'	'a6'	'a8'
-	0.18 ± 0.19	0.11 ± 0.16	0.10 ± 0.33	0.41 ± 1.12	0.23 ± 1.39	0.30 ± 0.75	0.07 ± 0.16
1	0.17 ± 0.20	13.88 ± 8.38	35.31 ± 5.26	5.83 ± 5.52	5.13 ± 5.06	13.26 ± 11.63	0.06 ± 0.14
2	45.82 ± 24.10	13.90 ± 8.30	33.68 ± 7.98	5.80 ± 5.37	5.16 ± 5.06	32.94 ± 22.15	13.83 ± 7.83
3	0.61 ± 0.71	0.25 ± 0.32	0.10 ± 0.33	0.43 ± 1.48	0.25 ± 1.81	0.31 ± 0.73	0.07 ± 0.16
4	0.18 ± 0.19	0.24 ± 0.19	0.10 ± 0.33	0.46 ± 1.18	0.31 ± 1.48	0.45 ± 0.81	0.26 ± 0.20
5	0.16 ± 0.18	0.20 ± 0.19	0.10 ± 0.33	0.47 ± 1.22	0.34 ± 1.49	0.44 ± 0.78	0.06 ± 0.14
6	0.16 ± 0.21	0.11 ± 0.17	0.10 ± 0.33	0.69 ± 1.54	0.40 ± 1.73	0.32 ± 0.76	0.07 ± 0.16
7	0.17 ± 0.19	0.11 ± 0.17	0.10 ± 0.33	0.69 ± 1.54	0.42 ± 1.73	0.32 ± 0.74	0.07 ± 0.16
8	0.18 ± 0.19	0.10 ± 0.15	0.10 ± 0.33	0.43 ± 1.14	0.23 ± 1.39	0.31 ± 0.71	0.07 ± 0.15
9	0.17 ± 0.19	0.10 ± 0.16	0.10 ± 0.33	0.43 ± 1.15	0.22 ± 1.41	0.31 ± 0.73	0.07 ± 0.16
1,2	45.24 ± 24.25	21.55 ± 7.85	51.48 ± 8.07	8.89 ± 6.56	7.90 ± 6.15	36.79 ± 23.66	13.71 ± 7.77
3,4	0.60 ± 0.71	0.36 ± 0.31	0.10 ± 0.33	0.48 ± 1.54	0.33 ± 1.92	0.46 ± 0.84	0.26 ± 0.19
3,4,5	0.69 ± 0.82	0.41 ± 0.30	0.10 ± 0.33	0.54 ± 1.65	0.43 ± 2.01	0.57 ± 0.91	0.25 ± 0.17

Table 4.5: Comparison of the best AAE obtained with the lLBA and the vLBA, respectively. The error values are written side by side and listed together with the relevant parameters. The minimal AAE of each sequence is highlighted in blue. In the case of the sequences 'a1' and 'a3', the results are the same, but the standard deviation of the vLBA is lower.

	lLBA				vLBA			
seq	ρ	t	K	AAE [°]	AAE [°]	ρ	t	K	λ
'a1'	21	7	4	0.131 ± 0.152	0.131 ± 0.150	21	7	4	0.0001
'a2'	21	7	9	0.082 ± 0.137	0.075 ± 0.118	21	7	9	0.0010
'a3'	5	7	2	0.058 ± 0.420	0.058 ± 0.378	5	7	2	0.0001
'a4'	21	7	18	0.212 ± 1.125	0.182 ± 1.474	15	7	25	0.0010
'a5'	11	7	22	0.194 ± 1.436	0.175 ± 1.547	11	7	22	0.0005
'a6'	15	7	8	0.305 ± 0.747	0.271 ± 0.756	11	7	8	0.0010
'a8'	21	7	3	0.047 ± 0.097	0.046 ± 0.094	21	7	3	0.0005

effect on the resulting AAE. Omitting all three models simultaneously, deteriorates the result almost by a factor of four. This finding fits to the RIC of 'a2'. The plot of the RIC, which is shown in Figure 4.13, indicates that the first five POD modes include nearly 100% of the total information of sequence 'a2'. An extreme case is sequence 'a3', which has only two important POD modes, namely the first and the second one. This can either be seen from Figure 4.13 or from Table 4.4, which show that, if the first two modes are included within the set of motion models the result is always the same, independent of whether or not other modes are included.

The lowest values of the AAE that were achieved with the lLBA and the vLBA are shown in Table 4.5 together with the important parameters. In order to obtain these optimal values, the parameters ρ and t as well as the number of used motion models K, and in case of the vLBA also the weight parameter λ of the smoothness term (cf. Section 4.1.4) were optimized for each sequence individually. To this end ρ and t denote the spatial and temporal dimensions of the

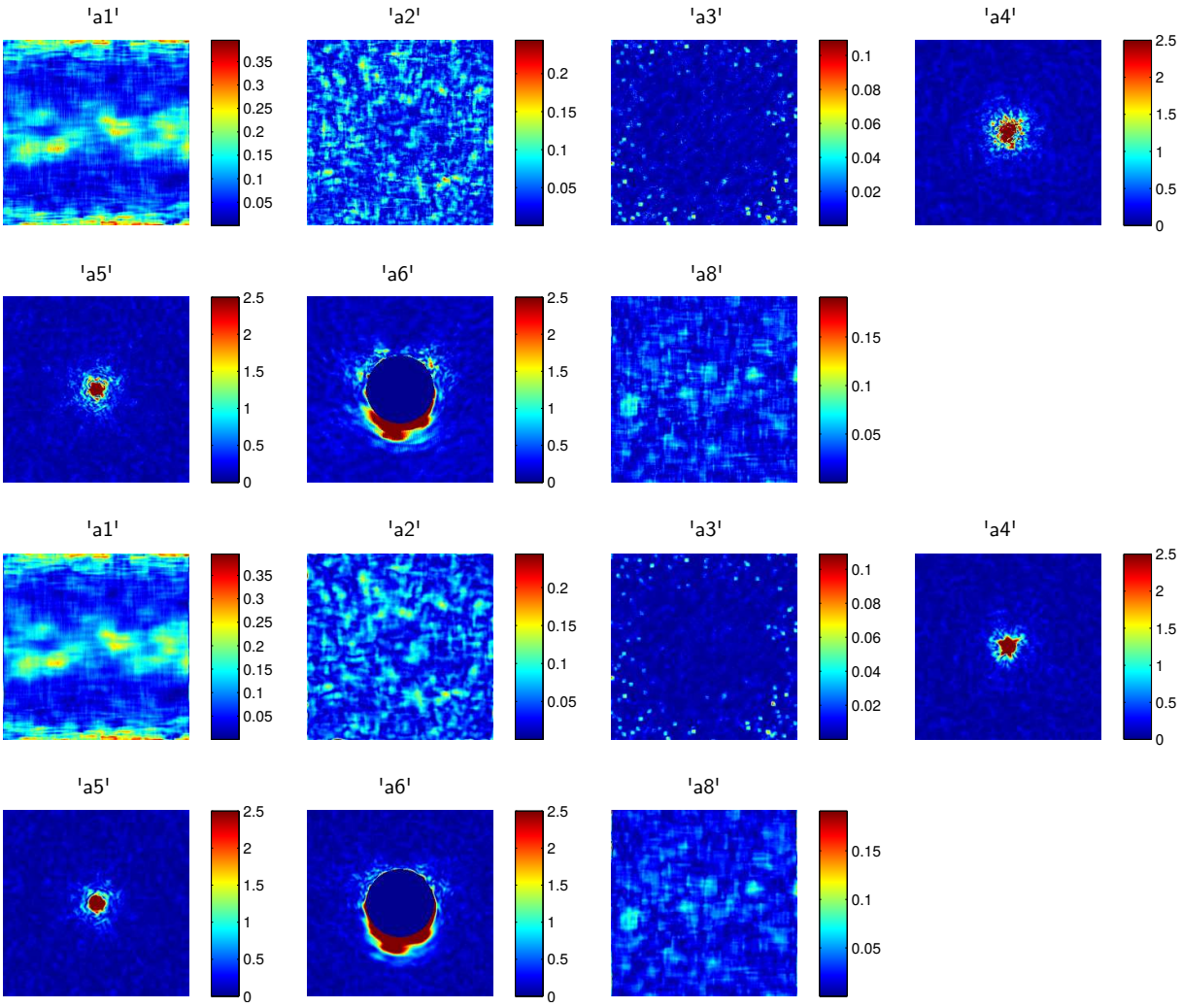

Figure 4.19: AE (color coded) of each sequence for the lLBA (*both top rows*) and the vLBA (*both bottom rows*). The parameters were chosen such that the lowest values of the AAE were obtained (cf. Table 4.5). Consider the different color ranges of the different sequences. For sequences 'a4', 'a5', and 'a6', the cutoff AAE value is $2.5°$, which means that values larger or equal $2.5°$ are displayed in the same color.

motion models. Therefore, the parameters were chosen from different sets: $\rho \in \{5, 11, 15, 21\}$, $t \in \{1, 3, 7\}$, $K \in \{2, \ldots, 25\}$, and $\lambda \in \{0.0001, 0.0005, 0.001, 0.005, 0.01, 0.02\}$. As explained in Section 4.1.4 a maximum number of 50 iteration steps and a SOR parameter $\omega = 1.8$ was used for the vLBA.

The Table 4.5 reveals that for most sequences the results of the vLBA are better than the results obtained with the lLBA. For the sequences 'a1' and 'a3', the values of the AAE are equal. Nevertheless, in these two cases the standard deviation of the errors obtained with the vLBA is lower. The largest difference between both approaches was obtained for the sequences 'a4', 'a5', and 'a6'. Overall, the vLBA appears to be the favorable method. This may be due to the filling-in effect introduced by the regularization term, which spreads information to image locations were the aperture problem is present.

Table 4.6: Influence of the utilized set of motion models on the estimated flow field. Each column of the table contains results in form of the AAE of the lLBA for one particular sequence obtained with different motion models. The motion models in the 'HS' row were learned from the Horn and Schunck flow field of the corresponding sequences. For each value, the spatial and temporal model size as well as the number of used motion models was optimized. All parameters are listed in Tabel 4.7. The minimal AAE of each sequence is highlighted in blue.

	AAE [°]						
model	'a1'	'a2'	'a3'	'a4'	'a5'	'a6'	'a8'
'a1'	0.13 ± 0.15	0.20 ± 0.18	0.05 ± 0.32	0.27 ± 1.87	0.27 ± 1.76	0.38 ± 0.81	0.06 ± 0.11
'a2'	0.11 ± 0.16	0.08 ± 0.14	0.05 ± 0.30	0.19 ± 1.80	0.16 ± 1.31	0.23 ± 0.68	0.07 ± 0.14
'a3'	0.55 ± 0.63	0.36 ± 0.28	0.06 ± 0.42	0.33 ± 1.94	0.34 ± 2.11	0.47 ± 0.82	0.23 ± 0.17
'a4'	0.66 ± 0.81	0.39 ± 0.46	0.16 ± 0.28	0.21 ± 1.12	0.29 ± 2.15	0.35 ± 0.77	0.31 ± 0.24
'a5'	0.28 ± 0.31	0.19 ± 0.22	0.08 ± 0.34	0.22 ± 1.57	0.19 ± 1.44	0.29 ± 0.77	0.14 ± 0.19
'a6'	0.14 ± 0.20	0.11 ± 0.16	0.08 ± 0.34	0.21 ± 1.79	0.17 ± 1.20	0.30 ± 0.75	0.09 ± 0.15
'a8'	0.16 ± 0.15	0.20 ± 0.18	0.06 ± 0.41	0.27 ± 1.87	0.27 ± 1.77	0.38 ± 0.81	0.05 ± 0.10
'mixed'	0.14 ± 0.20	0.11 ± 0.16	0.06 ± 0.42	0.20 ± 1.68	0.19 ± 1.23	0.27 ± 0.78	0.07 ± 0.15
'HS'	0.14 ± 0.15	0.09 ± 0.13	0.04 ± 0.23	0.17 ± 1.27	0.16 ± 1.33	0.26 ± 0.76	0.05 ± 0.12

Table 4.7: The optimal parameter values for the number of motion models $K \in \{2, \ldots, 25\}$, the spatial model size $\rho \in \{5, 11, 15, 21\}$, and the temporal model size $t \in \{1, 3, 7\}$ that yielded the AAE values listed in Table 4.6. Marked in blue are the parameters corresponding to the minimal error values of each sequence.

	'a1'			'a2'			'a3'			'a4'			'a5'			'a6'			'a8'		
model	K	ρ	t	K	ρ	t	K	ρ	t	K	ρ	t	K	ρ	t	K	ρ	t	K	ρ	t
'a1'	4	21	7	4	11	7	6	5	7	4	11	7	4	11	7	4	11	7	6	21	7
'a2'	6	21	7	9	21	7	13	5	7	5	11	7	20	21	7	12	21	7	9	21	7
'a3'	15	11	7	3	11	7	2	5	7	2	11	7	4	11	7	3	11	7	2	11	7
'a4'	16	11	7	25	15	7	23	5	7	18	21	7	25	15	7	25	15	7	25	15	7
'a5'	19	15	7	24	11	7	25	5	7	25	11	7	22	11	7	22	11	7	24	11	7
'a6'	4	15	7	5	15	7	6	5	7	5	11	7	5	15	7	9	15	7	4	15	7
'a8'	3	15	7	4	11	7	3	5	7	4	11	7	4	11	7	4	11	7	3	21	7
'mixed'	4	15	7	5	15	7	2	5	7	7	11	7	10	11	7	5	11	7	4	15	7
'HS'	4	21	7	9	21	7	13	5	7	9	15	7	9	15	7	5	11	7	4	21	7

Images of the AE corresponding to the AAE values and the respective parameters given in Table 4.5 are shown for the lLBA and the vLBA in Figure 4.19. In order to eliminate the error values introduced at the image boundaries by the convolution with the gradient filter kernels (cf. Section 2.3.4), the AE images were slightly cropped. For a better visibility, the range of the color bar of the sequences 'a4', 'a5', and 'a6' was restricted to the interval $[0, 2.5]$. AE values larger or equal 2.5° are displayed in the same color. A visual comparison of the corresponding images from both methods shows, that the images obtained with the vLBA are slightly smoother than the ones obtained with the lLBA. Small error spikes obtained by the local approach were slightly smoothed down by the global approach which is due to the additional smoothness constraint used within the vLBA.

4.3.5 Influence of the training data

So far only motion models learned from the own correct flow fields of each sequence were used to estimate the optical flow fields with the LBA. In this section, the effect of the utilized motion models on the achieved AAE and therewith the optical flow field is described. All eight sets of motion models displayed for $\omega = 11 \times 11 \times 1$ in Figure 4.11, were utilized to estimate the flow field of each sequence. In addition to the seven sets of motion models learned from the seven sequences, the mixed set was used. This means that for instance for the estimation of the flow field of sequence 'a1', the motion models obtained from 'a1' as well as the remaining sets of motion models were utilized. Furthermore, also motion models learned from flow fields previously determined with the standard Horn and Schuck optical flow approach [Horn and Schunck, 1981] described in Section 3.4.4, were employed. In this case, only the Horn and Schunck flow fields learned from each sequence were used to determine the optical flow field of this particular sequence.

The results in from of the AAE obtained with the lLBA are given in Table 4.6. For each combination, the parameter values yielding the lowest AAE were chosen. Each parameter selection is listed in Table 4.7. The AAE values obtained with the vLBA are generally lower than those obtained with the lLBA. The conclusion about the applicability of each set of motion models is the same for both methods and, therefore, these results are not shown here.

The results of each particular sequence obtained with the *own* motion models were always amongst the best, but they were mostly not the very best choice. Overall, the motion models learned from 'a2' yielded the lowest error values and, therefore, these motion models are perfectly suited for most of the analytic sequences. Only for 'a3', 'a4', and 'a8', the 'a2' motion models did not yield the lowest AAE, nevertheless, the results were second-best for these three cases. The motion models learned from 'a6', 'mixed', and 'HS' yielded also very good results for all sequences. It is not surprising that the motion models of the three sequences 'a2', 'a6' and 'mixed' yielded similar results, because the first few motion models look similar (cf. Figure 4.11). The 'HS' motion models are different for each sequence, because they were learned from each Horn and Schunck flow field of the particular sequence. For the sequences 'a3' and 'a4', these motion models yielded the lowest AAE.

On the other hand the motion models learned from 'a3' performed very poorly for all sequences apart from 'a3' itself. This may be due to the fact, that this set in essence contains only two models with constant motion in two perpendicular directions and, therefore, cannot accurately model advanced flow structures. The performance of the motion models learned from 'a1', 'a5', and especially from 'a4' was worse compared to the other models.

Table 4.7 indicates that if the models learned from 'a4' and 'a5' were used, a relatively large number of motion models was required in order to reach the lowest error value of each sequence that is possible with these sets of motion models. If the models learned from 'a3' were used, the number of used motion models was usually low, because this sequence contains only two expedient modes. For the motion models learned from the other sequences, K ranged between

Table 4.8: AAE obtained with the statistical lLBA. The parameters N_ρ, N_t, and the averaging method were chosen to yield the lowest error values. The last column contains the values from Table 4.5 obtained with the normal lLBA. The minimal AAE of each sequence is highlighted in blue.

							AAE [°]	
seq	ρ	t	K	method	N_ρ	N_t	slLBA	lLBA
'a1'	21	7	4	mean	11	1	0.119 ± 0.143	0.131 ± 0.152
'a2'	21	7	9	med	11	1	0.070 ± 0.135	0.082 ± 0.137
'a3'	5	7	2	med	5	1	0.040 ± 0.291	0.058 ± 0.420
'a4'	21	7	18	tmean	7	3	0.170 ± 1.165	0.212 ± 1.125
'a5'	11	7	22	tmean	3	7	0.186 ± 1.406	0.194 ± 1.436
'a6'	15	7	9	tmean	9	7	0.242 ± 0.730	0.305 ± 0.756
'a8'	21	7	3	med	11	1	0.043 ± 0.094	0.047 ± 0.097

Table 4.9: AAE obtained with the statistical vLBA. The parameters N_ρ, N_t, and the averaging method were chosen to yield the lowest error values. The last column contains the values from Table 4.5 obtained with the normal vLBA. The minimal AAE of each sequence is highlighted in blue.

								AAE [°]	
seq	ρ	t	K	λ	method	N_ρ	N_t	svLBA	vLBA
'a1'	21	7	4	0.0001	mean	11	1	0.120 ± 0.142	0.131 ± 0.150
'a2'	21	7	9	0.0010	med	11	1	0.069 ± 0.117	0.075 ± 0.118
'a3'	5	7	2	0.0001	med	5	1	0.047 ± 0.300	0.058 ± 0.378
'a4'	15	7	25	0.0010	mean	11	1	0.173 ± 1.451	0.182 ± 1.474
'a5'	11	7	22	0.0005	tmean	3	7	0.174 ± 1.527	0.175 ± 1.547
'a6'	11	7	8	0.0010	tmean	7	7	0.254 ± 0.760	0.271 ± 0.756
'a8'	21	7	3	0.0005	med	11	1	0.043 ± 0.092	0.046 ± 0.094

four and nine. The ideal number of motion models depended on each set of learned motion models and was related to the RIC of the particular set. The spatial dimension ranged from $\rho = 11$ to $\rho = 21$ with exception of sequence 'a3' where the ideal size was $\rho = 5$ for all sets of motion models. The ideal temporal dimension was in any case $t = 7$.

Table 4.6 shows that the results obtained with foreign motion models are comparable to, or in many cases even better than the results obtained with the *own* motion models. Therefore, it is possible to use existing vector fields of known flow problems with similar flow features than the one of interest to determine the optical flow field. The usage of motion models learned from flow fields, that were previously determined with alternative methods is also possible.

4.3.6 Statistical approach

In order to test the statistical LBA the flow fields of the analytic sets were estimated using the motion models learned from their own correct flow fields. The lowest achieved AAE values are shown in Table 4.8 and Table 4.9 for the *statistical local learning-based approach* (slLBA) and the *statistical variational learning-based approach* (svLBA), respectively. Thereby, the optimal parameter values for ρ, t, and K given in Table 4.5 were adopted. The parameters N_ρ and N_t denote the number of vector patches in each spatial and the temporal direction from which the

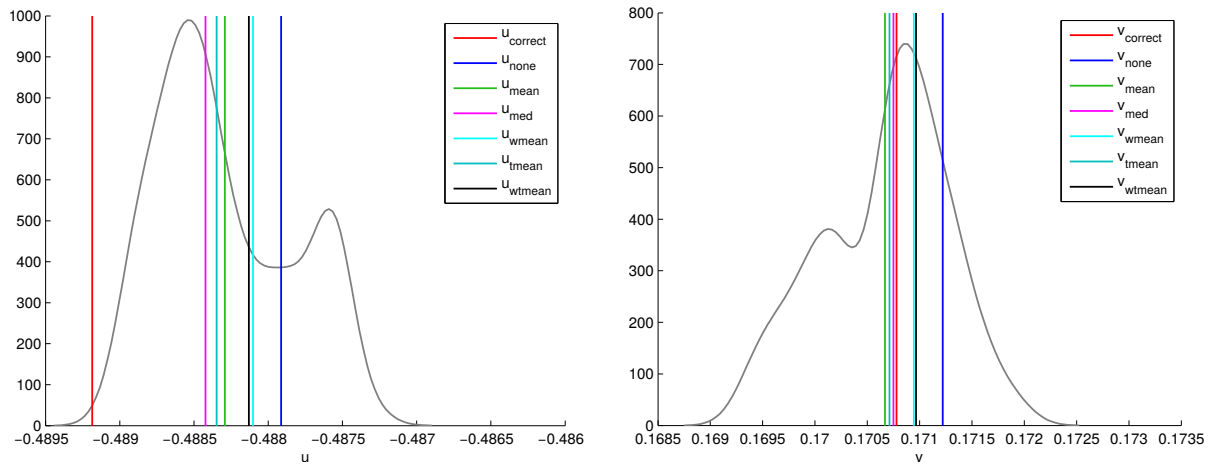

Figure 4.20: Kernel density estimations of the samples of the statistical ILBA at a particular location $(44, 99)$ of sequence 'a2'. *Left*: Horizontal vector components. *Right*: Vertical vector components. The colored lines show the displacement in pixels obtained with different averaging methods.

statistical sample data was taken. Therefore, they control the total number of flow estimates within the data sample, which is given by $2 \cdot N_\rho \cdot N_t$. In order to obtain the lowest AAE, the five averaging methods introduced in Section 4.1.6 were tested with different parameter choices $\rho_s \in \{3, 5, 7, 9, 11\}$ and $t_s \in \{1, 3, 7\}$. The following abbreviations are used in Table 4.8 and Table 4.9 for the method: mean (arithmetic mean), med (median), wmean (weighted mean), tmean (truncated mean), and wtmean (weighted and truncated mean).

For comparison of the statistical LBA and the normal LBA, additionally the AAE obtained with the latter one is shown in the last columns of Table 4.8 and Table 4.9. The statistical approaches led to greatly improved results. The preferred averaging methods were mean, median, and truncated mean. In neither case, the weighted mean or the weighted and truncated mean yielded the lowest AAE. With the exception of sequence 'a4', the local and the variational approaches favored the same averaging method. In all cases, the statistical methods yielded lower error values than their non-statistical counterparts. In average the AAE of each statistical version is approximately 15 % lower for the local approach and about 8 % lower for the variational approach.

Typical distributions of the displacements within the statistical sample data are displayed in the kernel density estimations shown in Figure 4.20. The displacements in horizontal direction are shown on the left and the displacements in vertical direction on the right. The sample data was taken from an arbitrarily chosen location (here $(44, 99)$) of sequence 'a2'. With parameters $N_\rho = 11$ and $N_t = 7$ the number of vectors within the sample was 847. The width of the distribution is only about 0.003 px in both cases. The difference between neighboring vectors of the correct vector field is in the same order of magnitude.

The kernel density estimation is a non-parametric way to estimate the probability density

Table 4.10: AAE of the rvLBA. For a better comparability to the other methods, the statistical and the normal vLBA are shown in the last two columns. The minimal AAE is highlighted in blue.

							AAE [°]	
seq	ρ	t	K	λ	ϵ	rvLBA	stat vLBA	vLBA
'a1'	21	7	4	0.0001	0.75	0.130 ± 0.150	0.120 ± 0.142	0.131 ± 0.150
'a2'	21	7	9	0.0010	0.65	0.075 ± 0.119	0.069 ± 0.117	0.075 ± 0.118
'a3'	5	7	2	0.0001	0.1	0.056 ± 0.407	0.047 ± 0.300	0.058 ± 0.378
'a4'	15	7	25	0.0010	1.5	0.170 ± 1.366	0.173 ± 1.451	0.182 ± 1.474
'a5'	11	7	22	0.0005	0.8	0.165 ± 1.443	0.174 ± 1.527	0.175 ± 1.547
'a6'	11	7	8	0.0010	1.1	0.266 ± 0.738	0.254 ± 0.760	0.271 ± 0.756
'a8'	21	7	3	0.0005	1.85	0.046 ± 0.094	0.043 ± 0.092	0.046 ± 0.094

function of the sample. The estimated probability density functions (Figure 4.20) are clearly not normal distributed. Therefore, the used averaging methods might differ from the expected value of the samples. Additionally to the distribution, the values of the correct displacements as well as the estimated displacements with and without averaging operation are marked by colored lines (cf. Figure 4.20). Out of all averaging methods, the median is closest to the correct displacement for both components. As indicated in Table 4.9 the median is also the desired averaging method for this sequence. All other methods including the estimate obtained without averaging, which corresponds to the normal LBA, yield displacements further away from the correct displacement. However, as for instance the case for the horizontal component shown on the left of Figure 4.20, the correct displacement may be located very much at the side of the distribution. Therefore, no value derived from the distribution can yield the correct displacement in this case. Nevertheless, as indicated in the Tables 4.8 and 4.9 the statistical approaches clearly improve the obtained results and provide advantages.

4.3.7 Robust variational approach

The results obtained with the rvLBA and the *own* motion models are shown in Table 4.10. As robust error function, the regularized TV penalty function defined in Equation (4.19) was used for the data as well as for the smoothness term. For simplicity reasons, the regularization parameter ϵ was set to the same value in both terms. The other parameters such as ρ, t, K, and λ were adopted from Table 4.5, where the lowest possible error values obtained with the lLBA and the vLBA are depicted. The number of outer and inner fixed point iterations was set to ten. However, in the case that a fixed point was reached with respect to a tolerance of 10^{-8} in lesser than ten iteration steps, the iteration was stopped successfully. For a comparison with the normal and the statistical vLBA, the error values of these approaches are also given in Table 4.10.

For the sequences 'a4' and 'a5', the rvLBA achieved the lowest AAE. Both sequences share in common, that they contain very large displacements in a small area in the central part of the images. At these locations, the AE is particularly large, which is due to the linearized

BCCE, wrong particle pair assignment, particle losses, and extreme gradients. Therefore, in these cases the use of robust error functions was advisable and led to improved results. For all other sequences, the lowest error values were obtained with the svLBA. Nevertheless, the error values obtained with the rvLBA are never higher than the error values achieved with the vLBA. For the sequences 'a1', 'a3', 'a4', 'a5', and 'a6' they are in fact lower. This indicates an accuracy gain of the rvLBA compared to the vLBA. On average, the AAE of the rvLBA is approximately 3 % lower than the AAE of the vLBA.

Due to the absence of noise in the analytic test set, these sequences may not be an ideal test cases for the rvLBA. With the exception of the object boarder in sequence 'a6', there are also no flow discontinuities in the sequences, which require the usage of a flow-driven approach such as the rvLBA. However, because the object area in sequence 'a6' was masked, the non-robust approaches were also applicative and yielded good results. Especially, the statistical approach with the truncated mean led to a low AAE.

4.3.8 Computation time

In order to analyze the efficiency of the LBA, the calculation time was monitored. In Figure 4.21 the computation time per pixel is shown for different spatio-temporal dimensions and different numbers of motion models for the local (lLBA), the variational (vLBA), and the robust (rvLBA) approach. The results were computed on the 'a2' sequence of the analytic set. Within the vLBA, 50 iteration steps were executed and the rvLBA was used with 5 inner and 5 outer fixed point iterations. All computations were conducted on a 2.80 GHz Intel® Core™ i7-860 CPU executing MATLAB code. Because the set of learned motion models needs to be calculated only once and can be used for different estimations of the optical flow field, the computation of the POD modes is not considered for the determination of the computation time.

With increasing number of used motion models, the computation time increases approximately linearly for the lLBA (Figure 4.21). Whereas for the vLBA and the rvLBA the increase in computation time with a growing number of motion models is rather quadratic. For $K = 10$, the computation time rises from 0.26 ms for $\omega = 5 \times 5 \times 1$ to 1.78 ms for $\omega = 21 \times 21 \times 7$ for the lLBA. For the same increase of the spatio-temporal size, the computation time rises from 0.92 ms to 1.29 ms for the vLBA and from 1.48 ms to 1.87 ms for the rvLBA. This shows, that the computation time of the lLBA increases faster with a growing spatio-temporal dimension compared to both other approaches. However, the computation time of the lLBA is generally lower. For the statistical LBA, the computation time rises for all methods according to the chosen statistical sample size and the averaging method. The extra time, which has to be added to the times shown in Figure 4.21, ranges from 0.2 ms up to 0.8 ms per pixel depending on the size of the statistical sample.

The computation times shown in Figure 4.21 were obtained without a special optimization of the code. The focus within this project was rather on the functionality of the method than on the computation speed. The shown computation times give only a rough survey of the relative

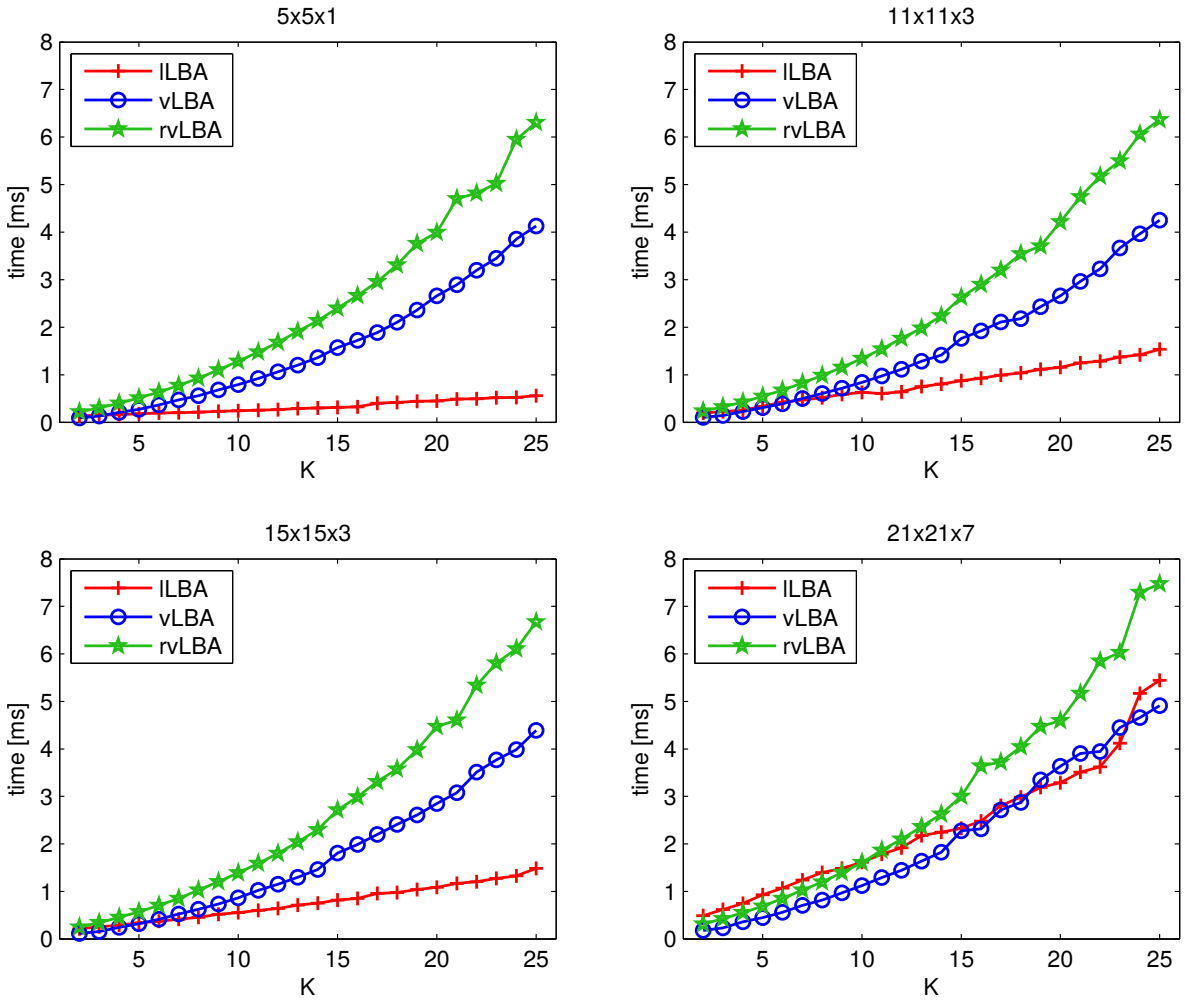

Figure 4.21: Comparison of the computation time of the different versions of the LBA for different dimensions and numbers of used motion models.

performances of the different approaches and their dependency on size and number of motion models. Therefore, no comparison of the LBA with the computation times of other optical flow methods has been performed in this thesis.

Nevertheless, there are several possibilities to achieve performance gains. First of all, the local version (lLBA) can be easily parallelized. This is possible, because the determination of one individual flow vector is independent of the other flow vectors. Accordingly, it could be implemented in graphics processing units (GPUs) following the description of Strzodka and Garbe [2004] who used such a technique to achieve real-time performance. A similar performance should be possible for the lLBA. In order to improve the performance of the variational and the robust variational approach, multigrid methods, could be used to speed up the computation. These methods belong to the fastest numerical schemes for solving systems of equations [Briggs et al., 2000]. The problem is solved hierarchically on various scales to obtain initializations that are already very close to the correct solution. This procedure drastically reduces the required

iteration steps. As described by Bruhn et al. [2006], using so called *bidirectional* multigrid methods, real time performance is possible. A corresponding implementation of the vLBA and the rvLBA should lead to a great performance gain.

4.4 Conclusion

All of the different versions of the LBA performed very well on the synthetic test sequences and the conducted tests lead to the following conclusion. The ideal number and size of the applied typical motion models depends on the complexity of the studied fluid flow. Simple flows such as the uniform motion in sequence 'a3' require only two motion models of relatively small size (e.g., $5 \times 5 \times 7$). However, the ideal number and size increase with increasing complexity of the studied flows. The inclusion of temporal information by using spatio-temporal motion models with $t > 1$ leads to more accurate flow estimates. In order to obtain suitable motion models the training data should contain similar flow structures than the flow field under study. Omitting selected important motion models leads to deteriorated flow fields.

A Comparison of the results obtained with different versions of the LBA on the test sequences showed, that the variational approach performed better than the local one. The AAE obtained with the vLBA was approximately 7 % lower than the AAE obtained with the lLBA. Moreover, the AE obtained with the vLBA was smoother than the AE obtained with the lLBA. This is mainly due to the additional smoothness constraint, which is applied within the variational approach. An additional performance gain could be obtained by using the statistical versions of the LBA. They yielded lower error values on all seven test sequences. In average the AAE of the slLBA was 15 % lower than the AAE of the lLBA and the AAE of the svLBA was 8 % lower than the AAE of the vLBA. The rvLBA, which uses non-quadratic error norms for the data and the regularization term, also led to lower error values compared to the vLBA with quadratic error norms. The results improved on average by 3 %. Thus, the improvement was not as distinct as the improvement obtained with the svLBA. The computation time of all methods increased with increasing number of motion models and increasing model size. If computation time is key, it can be reduced significantly by using parallelized code and hierarchical multigrid methods in case of the local and the variational approach, respectively.

5 Fluid dynamical applications

In this chapter the LBA, which was introduced and tested in Chapter 4, is applied to different fluid dynamical problems. The applicability of the method is examined and the results are compared to the results of established methods. In Section 5.1 the properties of particle image sequences are investigated and the ideal image characteristics for which the LBA works best are determined and compared to the conditions of PIV (cf. Section 3.3), which is the most common method used to estimate the velocity field from particle images. In Section 5.2 the performance of the LBA on different particle image sequences is investigated and the results are compared to the results of other common estimation techniques. Due to the large number of abbreviations it is once more referred to page 121 where all acronyms are listed.

5.1 Image characteristics

The properties of particle image sequences are studied in order to identify the conditions under which the LBA works best. Therefore, especially the effect of the *particle number density* (PND) as well as the amount of the particle displacement from frame to frame is investigated.

5.1.1 Influence of the particle number density

In order to study the influence of the PND on the quality of the estimated displacement field, the Lamb-Oseen vortex flow of sequence 'a2' of the analytic set (cf. Section 4.3.1) was used as test case. To obtain different PND, several image sequences containing various numbers of synthetic particles, were generated. Therefore, a gray value image was constructed by placing images of synthetic particles at random image locations. The number of locations depended on the desired PND. To generate the particles, Gaussian blobs of varying height, dimension, and standard deviation were constructed. Additionally, small random numbers were added to each pixel of the particles to create particles that are unique in their appearance. Finally, the sequence was obtained by repeated warping of the generated image with the Lamb-Oseen flow field. Practically, this was done by interpolating the gray values at shifted positions of the image to determine the gray value at the grid positions of the subsequent image. The shift was defined by the flow field. The PND of the constructed sequences ranged from $\rho_p = 0.01\,\text{ppp}$ (particles per pixel) to $\rho_p = 0.35\,\text{ppp}$.

In Figure 5.1 the influence of the PND on the AAE is shown for the lLBA, the *Horn and Schunck approach* (HSA) [Horn and Schunck, 1981], and PIV. For the lLBA, the first nine

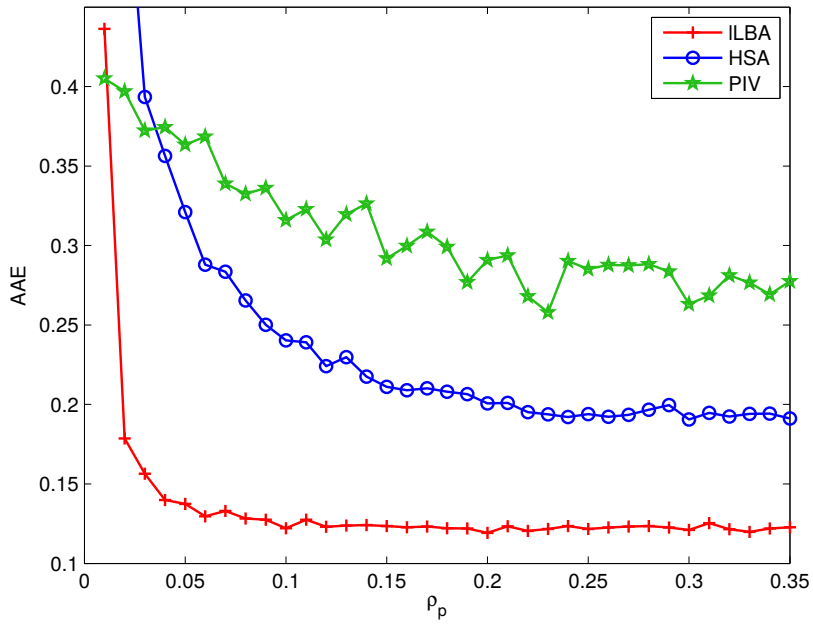

Figure 5.1: Influence of the PND on the AAE for the lLBA, the standard Horn and Schunck optical flow approach (HSA), and PIV.

motion models of size $21 \times 21 \times 7$, which were learned from the correct flow field, were used. The HSA, which is the standard global optical flow approach, was applied with different regularization parameters, which were optimized for each PND. PIV was conducted using the fluere 1.3 software package (cf. Section 3.3.3) with a final interrogation window size of 32×32. The AAE obtained with the lLBA decreases relatively fast for increasing PND and stays constant for $\rho_p > 0.1$ ppp. Therefore, the ideal PND for the application of the LBA is $\rho_p > 0.1$ ppp. The HSA shows also a strong decrease of the AAE followed by a plateau for $\rho_p > 0.25$ ppp. The decrease of the AAE of PIV is less steep but a clear trend towards smaller errors for increased PND is visible. This is in agreement with the findings of Raffel et al. [2007] where it is shown that a higher particle number leads to a higher accuracy, which is due to higher particle pair detection rates as well as lower measurement uncertainties. For PND smaller than 0.01 ppp, optical flow methods fail, whereas PIV still manages to estimate the flow field. A rule of thumb is that PIV applications require at least eight particles per interrogation window in order to ensure a detection probability of 95 % [Raffel et al., 2007]. One should keep in mind, that the above findings are based on synthetically generated images. The results may look different for real particle images and varying particle diameters.

5.1.2 Influence of the displacement

Due to the linearization of the BCCE (cf. Section 3.4.1) as well as the use of gradient images, optical flow approaches are not suitable to handle displacements that are larger than approximately one or two pixels. According to Liu and Shen [2008], optical flow methods require such a small displacement that the corresponding particles in two consecutive frames remain in contact. Otherwise the gradient images with respect to time cannot be accurately calculated and, thus,

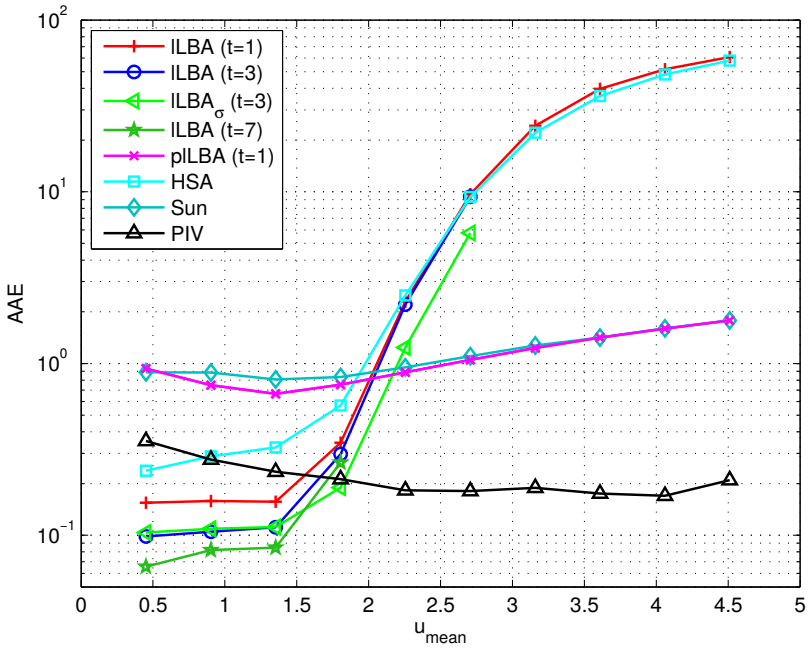

Figure 5.2: AAE of different optical flow methods and PIV in dependence of the mean velocity of sequence 'a2' from the analytic set.

optical flow methods fail. However, hierarchical multi-scale methods, which were introduced in Section 3.4.6, are able to compensate this deficit to a certain extent. For a thorough inspection of this fact, the influence of the displacement on the performance of different methods was investigated. Therefore, the Lamb-Oseen vortex flow of sequence 'a2' of the analytic set introduced in Section 4.3.1 was once more considered as test case. The sequence consists of 41 frames as well as the correct displacement field. As stated in Table 4.1 the mean displacement of the whole vector field is 0.45 px. By leaving away some frames, different image sequences with different mean displacements were generated. For instance, if only every second frame is considered, the mean displacement rises up to 0.9 px. The reference flow field was constructed by integrating the correct flow fields of the omitted images.

In Figure 5.2 the AAE of different methods in dependance of the mean displacement $\boldsymbol{u}_{mean}$ is plotted with logarithmic scale on the ordinate. The figure contains different versions of the lLBA with spatial size $\rho = 15$ and different temporal dimensions $t \in \{1, 3, 7\}$. The number of ideal motion models was optimized for each version. Also included in the plot is a pyramidal multi-scale version (plLBA). The method lLBA$_\sigma$ equals method lLBA but additionally Gaussian presmoothing with standard deviation $\sigma = 0.5$ was applied. Two other shown optical flow approaches are, the standard HSA [Horn and Schunck, 1981] and the *classical* method introduced by Sun et al. [2014], which applies a multi-resolution method similar to the plLBA. Additionally, results obtained with the PIV software fluere 1.3 (cf. Section 3.3.3) with a final interrogation window size of 32×32, are shown. The spatio-temporal versions of the lLBA require more than a pair of images in order to estimate the flow field and, therefore, the maximal possible mean displacement was smaller than for purely spatial approaches, since the original image sequence contains only 41 frames.

The plots of the optical flow approaches without multi-resolution techniques show a fast increase of the AAE starting with a mean displacement of approximately 1.5 px. For mean displacements smaller than 1.5 px, the error values of these methods are lower than the values of the other approaches. The AAE of the optical flow methods with multi-resolution techniques increase only slightly for mean displacements larger than 1.5 px. However, for small displacements the AAE is up to one order of magnitude higher than the AAE of the lLBA with $t = 7$. The AAE of the PIV approach decreases slightly with growing mean displacement. For displacements larger than 2, the AAE stays approximately constant. If preprocessing in form of Gaussian smoothing was applied (lLBA_σ), the AAE for displacements larger than 1.5 px is smaller than without preprocessing. Due to the Gaussian smoothing, distinct particles are isotropically diffused and originally separated particles in subsequent images become connected, whereas the displacements remain unchanged [Liu and Shen, 2008]. This leads to more accurate temporal derivatives and, thus, lower error values.

For optical flow approaches such as the LBA, the displacement between two frames of a particle image sequence should ideally be below two pixels. However, the common displacement of particle images recorded for correlation-based PIV methods is around 8 px [Raffel et al., 2007].

5.1.3 Optical flow versus particle image velocimetry

In order to estimate fluid flows from particle images, correlation-based PIV is the most common technique. However, optical flow methods specially tuned for the requirements of fluid flows and particle images also yield excellent results [Corpetti et al., 2006; Heitz et al., 2010; Ruhnau et al., 2005]. Nevertheless, as shown in the previous sections the requirements for both methods are slightly different. Whereas PIV prefers relatively large displacements, the ideal displacement for optical flow methods is rather small (< 1.5 px). The spatial resolution of PIV is mainly restricted by the interrogation window size since one window produces one flow estimate. This means that flow structures, which are smaller than the windows, cannot be considered. Especially in turbulent flow fields the size of the smallest measurable eddies is strongly limited. By using overlapping interrogation windows, the number of flow estimates can be increased.

An advantage of variational optical flow methods is that they produce dense flow fields with one flow estimate per pixel. However, the spatial resolution is not defined by the pixel grid. In purely global variational methods the resolution is reduced due to smoothing operations induced by the regularization term. In local global methods such as the vLBA the resolution is reduced on the one hand due to the regularization term and on the other hand by local averaging. Therefore, the exact spatial resolution depends also on the number and the size of the utilized motion models.

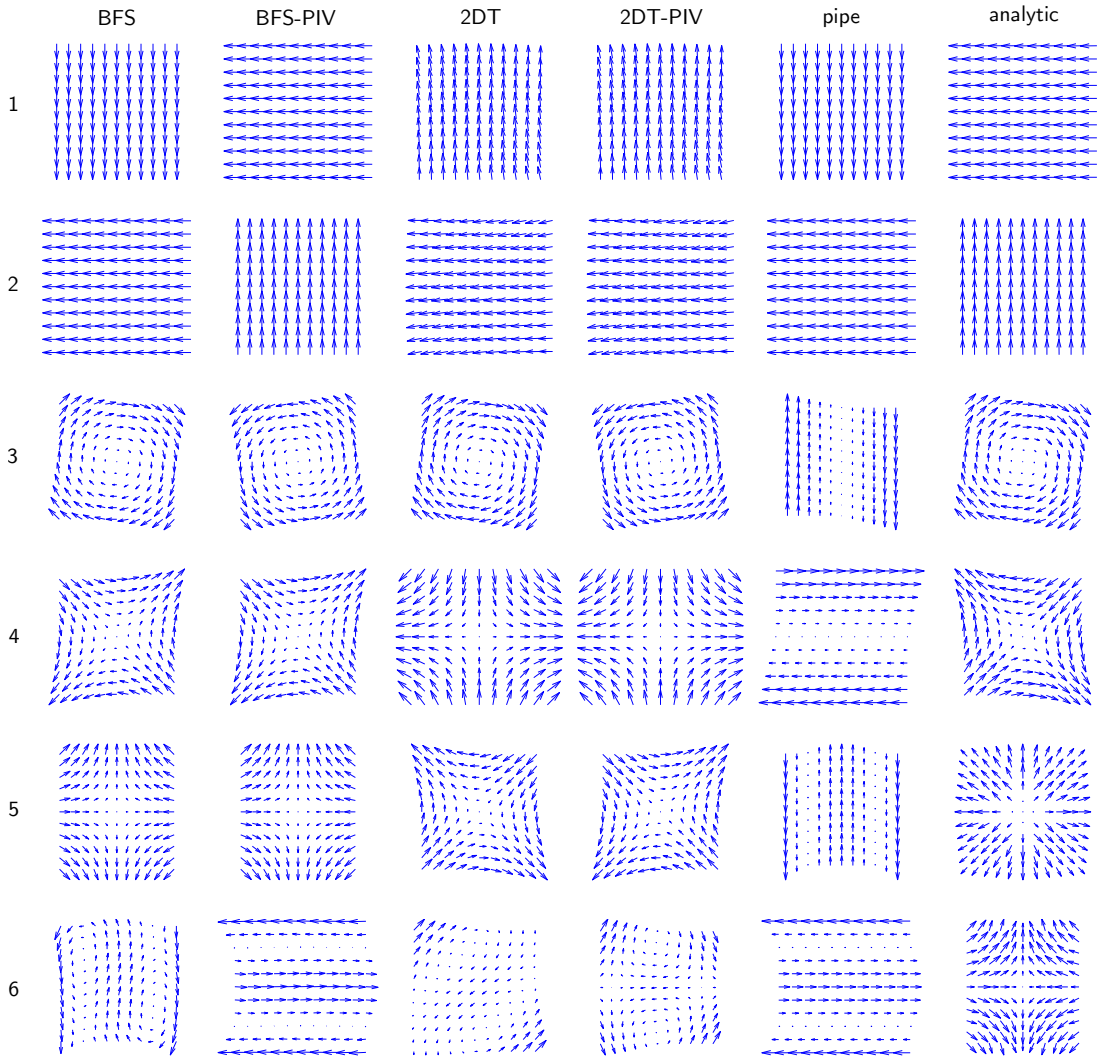

Figure 5.3: The first six motion models of size $11 \times 11 \times 1$ learned from the synthetic BFS and the 2D turbulent sequences as well as from a sequence containing a laminar pipe flow and from a combination of the different analytic sequences.

5.2 Applications and methods

In the following the performance of the LBA on different fluid dynamical test cases is presented. As test cases synthetic and real image sequences of the flow over a BFS as well as a laminar separation bubble flow and a 2D turbulent flow were considered. The motion models utilized for the BFS and the 2D turbulent sequences are shown in Figure 5.3. The figure contains the first six motion models of size $11 \times 11 \times 1$ learned from the synthetic BFS sequence, the 2D turbulent sequence, a sequence of a laminar pipe flow as well as from a combination of the seven analytic sequences introduced in Section 4.3.1. Thereby, 'BFS' and '2DT' denote the sets of motion models learned from the correct flow fields of the BFS and the 2D turbulent sequences, respectively. 'BFS-PIV' and '2DT-PIV' denote the sets of motion models learned

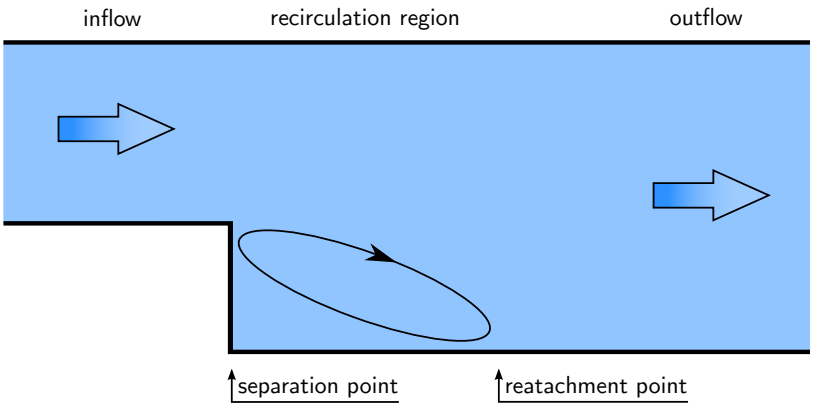

Figure 5.4: Sketch of a backward facing step (BFS) together with relevant flow properties.

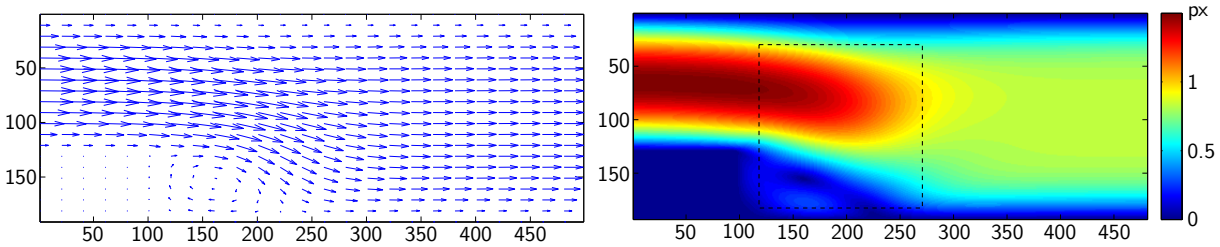

Figure 5.5: Simulated flow over a BFS. *Left*: Vector plot of the correct flow field. *Right*: Color coded magnitude of the displacements. The dashed square in the right image marks the area where the AE of different LBA methods is compared (cf. Figure 5.8).

from flow fields, which were previously estimated with correlation-based PIV (cf. Section 3.3). The motion models learned from the laminar pipe flow and the analytic sequences are denoted by 'pipe' and 'analytic', respectively. Within the scope of this thesis, some results of this section have already been published in Stapf and Garbe [2014a,b].

5.2.1 Synthetic backward facing step

The sequence

The flow over a BFS was considered to be a valuable fluid dynamical test case for the LBA because it contains some interesting flow aspects such as a sharp separation with a steep velocity profile in the free mixing layer and large rotations in the recirculation area. Furthermore the BFS is a traditional test case for experimental techniques of fluid flow measurements as well as for numerical simulations and, therefore, well understood. Due to its simple geometry and its easy reproducibility it is often employed for all kinds of methods. In Figure 5.4 a sketch of a BFS is shown. The flow is separated at the step edge and reattached again at the reattachment point further downstream. In the area behind the step, between the separation and the reattachment point, one or more recirculation eddies are formed by the flow. The reattachment length depends on the Reynolds number.

The flow field over the BFS was simulated in 2D with the finite element software library deal.II [Bangerth et al., 2007]. In Figure 5.5 a simulated flow field with Reynolds number $Re = 1000$

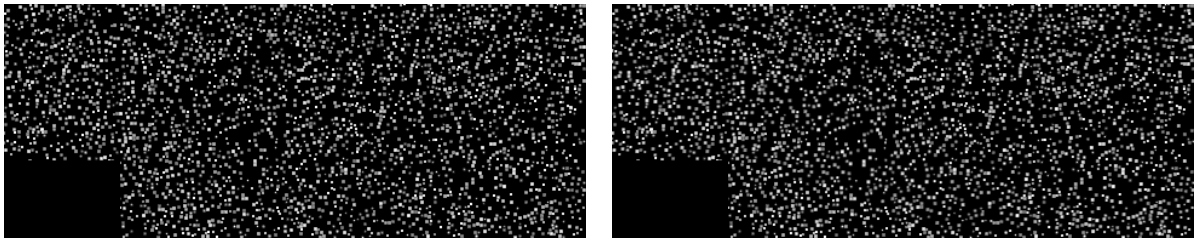

Figure 5.6: Two subsequent frames of the synthetically generated BFS image sequence.

and maximum velocity $v_{max} = 1.5\ \frac{\text{px}}{\text{frame}}$ is depicted. The left image contains a vector plot. For illustrative reasons, in vertical direction only every 10th and in horizontal direction only every 20th vector arrow is depicted. The right image contains the color coded magnitude values of the vectors, which represent the frame to frame displacements in pixels. In the vector plot the recirculation area behind the step, which contains a curl rotating in clockwise direction, is clearly visible.

The synthetic image sequence was generated by warping a gray value pattern with the simulated velocity fields. It was intended to be similar to the image sequences recorded in PIV measurements. Therefore, the gray value pattern consisted of synthetically created randomly spread tracer particles. In order to construct the first frame of the sequence, approximately 3000 particle locations were chosen randomly, and at each location an artificial particle was placed. The particles were created with a 2D Gaussian function and vary slightly in shape, size, and intensity. The resulting images are of size 481×193 px^2 and the PND approximately amounts to 0.035 particles per pixel (ppp). As shown in Section 5.1.1 increasing the PND leads to improved results of the LBA. Two subsequent frames of the synthetic image sequence are shown in Figure 5.6.

By adding Gaussian noise with zero mean and varying standard deviation σ several image sequences with different noise levels were created. The noise level is defined by the *noise-to-signal ratio* (NSR). It is given by the ratio of the standard deviation of the noise σ and the signal mean μ, that is $\text{NSR} = \frac{\sigma}{\mu}$. In order to generate sequences with different NSR, the standard deviation was enlarged in steps of 5 % from 0 % to 100 % of the signal mean.

The NSR is simply the inverse of the more common signal-to-noise ratio. Considering an ideal image sensor with only *shot noise*, which is due to a varying number of photons registered by the sensor at a given exposure level, the signal-to-noise ratio equals the square root of the signal mean [EMVA, 2010]. However, real sensors introduce additional noise, which can mainly be modeled by a normal distribution. Therefore, the signal-to-noise ratio is sensor specific.

Results

The flow field of the synthetic BFS sequence was estimated with the different versions of the LBA that were introduced in Chapter 4. Additionally, the influence of the applied training data

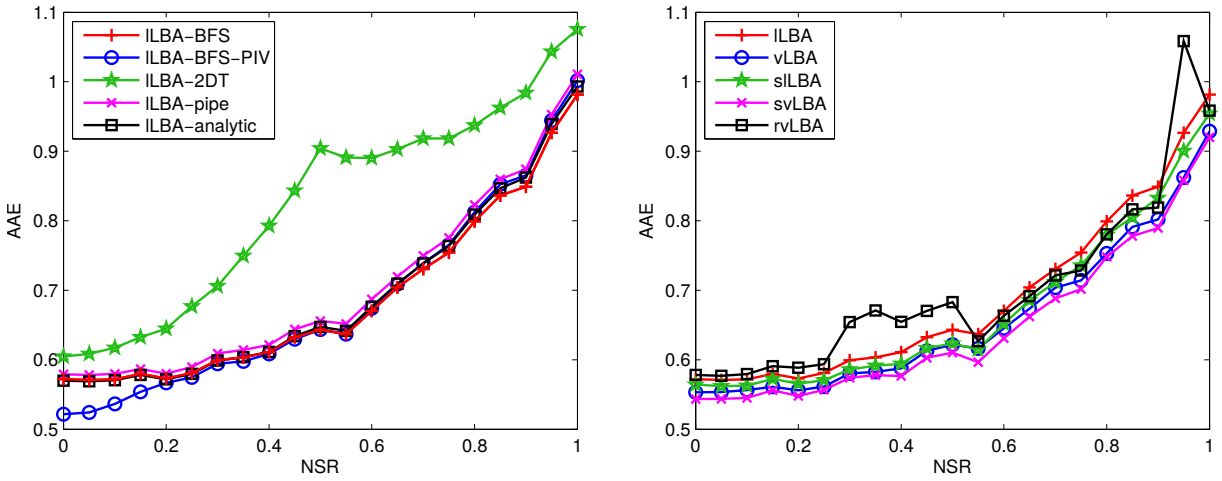

Figure 5.7: AAE of the LBA in dependence of the noise level given by the NSR. *Left*: Results obtained with the lLBA with different training data. *Right*: Results obtained with different versions of the LBA.

was observed as it was done for the analytic sequences in Section 4.3.5. The optimal size of the learned motion models was $15 \times 15 \times 7$. Previously to the estimation of the optical flow the images of the sequence were smoothed with a Gaussian kernel K_σ of standard deviation σ_ρ in spatial and σ_t in temporal direction. All relevant parameters have been optimized for each realization of the LBA. Besides σ_ρ and σ_t, this was basically the number of used motion models (K) as well as for the variational approaches vLBA and rvLBA the smoothness parameter λ and for the robust version also the regularization parameter ϵ of the penalty function defined in Equation (4.19). For the statistical versions namely the slLBA and the svLBA, the most capable averaging method was chosen for each realization.

The left plot of Figure 5.7 contains the results obtained with the lLBA using different sets of motion models. Thereby, the motion models displayed in Figure 5.3 were utilized with exception of '2DT-PIV'. The plot shows the dependency of the AAE on the NSR. The results are comparable for all motion models but the ones learned from the 2D turbulent sequence ('2DT'). In general, the error values increase with increasing noise level. For small NSR, the motion models learned from flow fields, which were previously estimated with PIV ('BFS-PIV'), yielded the lowest error values. With increasing noise level the motion models learned from the correct flow fields ('BFS') led to the best results. However, the difference of the AAE achieved with different sets of motion models was only marginal and with exception of the '2DT' models, any set of motion models yielded similar error values.

The reason for the poor performance of the '2DT' motion models compared to the other motion models is mainly due to variations within the first two models. Whereas the first two motion models of the other sets contain more or less constant motion in vertical and horizontal direction with parallel flow vectors of equal length, the first two models of the '2DT' set contain vectors that vary slightly in direction and length. Since, large parts of the BFS flow field constitute approximately constant flow, which cannot be modeled accurately with the '2DT'

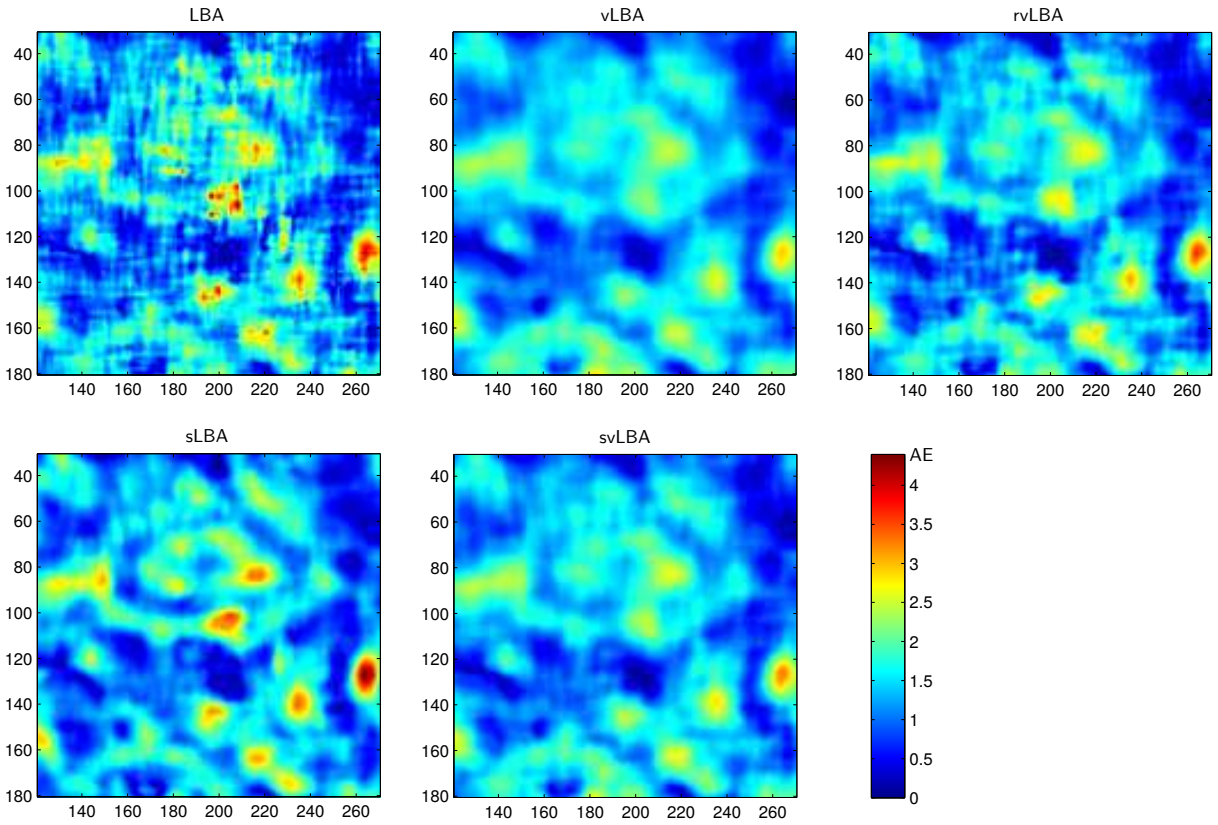

Figure 5.8: AE obtained with different LBA methods in the area marked by the dashed square in Figure 5.5 for the sequence with NSR $= 1$.

motion models, the AAE is larger than for the other motion models.

The right plot of Figure 5.7 contains the AAE in dependence of the NSR obtained with different versions of the LBA. Thereby, the models learned from the correct flow fields (BFS) were applied. The plot indicates a performance gain of the statistical versions of the LBA compared to their non-statistical counterparts. For all NSR, the svLBA yielded the lowest AAE. The error values obtained with the vLBA are slightly lower than the error values achieved with the lLBA. Especially for larger NSR, the results obtained with the rvLBA are comparable to the results obtained with the other methods. But overall, the rvLBA was not beneficial for this sequence, since it performed worse than the vLBA. In this case the application of quadratic error norms led to lower error values than the application of robust error norms.

In Figure 5.8 the AE values of the different LBA methods are compared visually for NSR $= 1$. Therefore, the AE is displayed color coded in the area marked in Figure 5.5. The error values obtained with the vLBA and the rvLBA are smoother than the error values obtained with the lLBA. This is mainly due to the applied smoothness constraint. Large error spikes, which are present in the lLBA image, were smoothed out by the variational approaches leading to smaller errors. Therefore, information is interpolated from the surrounding to areas where the estimation of the optical flow is somewhat problematic, e.g., due to a lack of gray value structures.

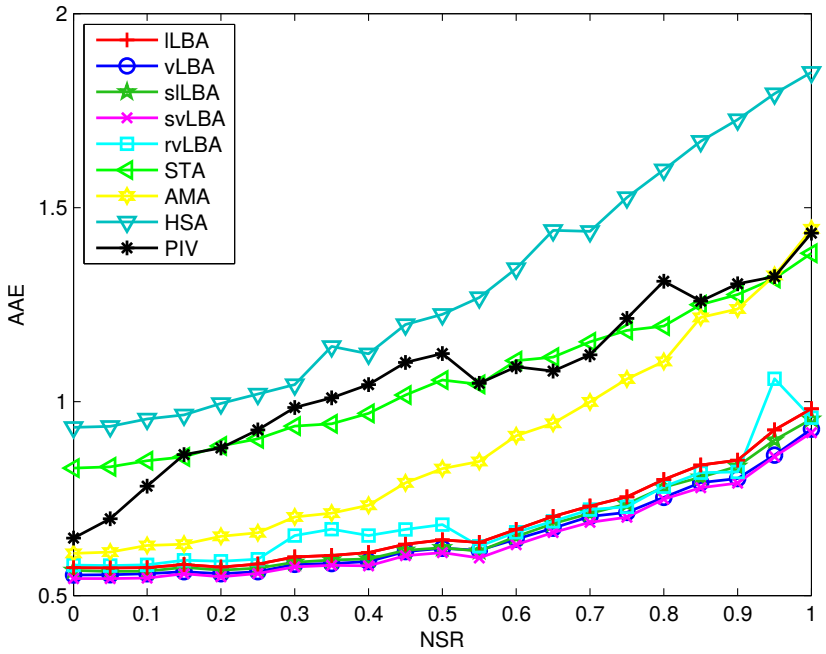

Figure 5.9: Comparison of different optical flow techniques on the BFS sequence, including the LBA, the STA, the AMA, the HSA as well as PIV.

As expected, also the AE images of the statistical approaches look smoother than the AE images of the non-statistical versions, which is mainly due to the applied averaging operation.

All versions of the LBA perform better than other tested methods, as can be seen in Figure 5.9. In this figure, the different versions of the LBA are compared to other optical flow methods as well as to correlation-based PIV. All methods were described in Chapter 3. The used optical flow methods were in essence, the local *structure tensor approach* (STA) [Bigün et al., 1991] with two parameters, the *affine model approach* (AMA) [Haussecker and Spies, 1999] with a six parameter model as well as the global HSA [Horn and Schunck, 1981]. All optical flow methods were implemented in MATLAB, whereas PIV was conducted with the software fluere 1.3 (cf. Section 3.3.3). In order to achieve the best results with each method, all parameters were optimized. PIV was performed with different interrogation window sizes and besides the standard correlation method also an ensemble correlation method was applied in order to include temporal information. From all variations always the one that yielded the lowest error values was selected.

Additionally to the plot in Figure 5.9, the results in form of the AAE as well as the ADE in x and y direction are presented in Table 5.1 together with each parameter choice for the three noise ratios NSR $= 0$, NSR $= 0.5$, and NSR $= 1$. The parameter K_σ denotes the size of the Gaussian kernel, which is used as weighting function by the two parametric approaches STA and AMA. Figure 5.9 and Table 5.1 show, that the LBA outperformed all other methods. In any case, the best results were obtained with the svLBA. In average the AAE achieved with this approach is approximately 25 % lower than the AAE of the second best method.

At least for low noise ratios the AAE of the AMA is comparable to the AAE of the LBA methods. Similar to the LBA this method is a parametric optical flow approach, which uses

Table 5.1: Performance of different methods on the synthetic BFS sequence. All parameters have been optimized for each method to yield the lowest possible error measures. Highlighted in blue is the lowest AAE of each noise level.

NSR	method	ρ	t	K	other parameters	σ_ρ	σ_t	AAE [°]	ADE_x $[px]$	ADE_y $[px]$
	LBA	15	7	5		1.5	0.0	0.572 ± 0.408	0.0140	0.0101
	vLBA	15	7	9	$\lambda = 0.01$	1.0	0.0	0.553 ± 0.340	0.0128	0.0096
	rvLBA	15	7	9	$\lambda = 0.005$, $\epsilon = 0.3$	1.0	0.0	0.578 ± 0.424	0.0137	0.0104
	sLBA	15	7	5	'mean'	1.5	0.0	0.564 ± 0.405	0.0134	0.0107
0.0	svLBA	15	7	9	$\lambda = 0.005$, 'med'	1.5	0.0	0.543 ± 0.362	0.0128	0.0099
	STA	11	7	2	$K_\sigma = 2.2$	0.8	0.0	0.828 ± 0.833	0.0234	0.0125
	AMA	11	7	6	$K_\sigma = 4.5$	1.0	0.0	0.603 ± 0.788	0.0199	0.0125
	HSA				$\lambda = 0.26$	0.8	0.4	0.933 ± 0.860	0.0248	0.0167
	PIV	16	1			1.5	0.0	0.648 ± 0.509	0.0184	0.0093
	LBA	15	7	5		1.1	0.0	0.643 ± 0.425	0.0162	0.0113
	vLBA	15	7	9	$\lambda = 0.01$	1.0	0.0	0.621 ± 0.392	0.0161	0.0112
	rvLBA	15	7	9	$\lambda = 0.008$, $\epsilon = 0.3$	1.0	0.0	0.683 ± 0.453	0.0195	0.0122
	sLBA	15	7	5	'tmean'	1.1	0.0	0.623 ± 0.402	0.0155	0.0113
0.5	svLBA	15	7	9	$\lambda = 0.01$, 'med'	1.1	0.0	0.610 ± 0.385	0.0158	0.0110
	STA	11	7	2	$K_\sigma = 3.7$	0.5	0.3	1.056 ± 1.739	0.4969	0.4030
	AMA	15	7	6	$K_\sigma = 6.5$	0.8	0.4	0.827 ± 1.118	0.0293	0.0165
	HSA				$\lambda = 0.5$	0.5	0.4	1.224 ± 1.142	0.0343	0.0232
	PIV	32	3			1.6	0.0	1.124 ± 2.284	0.0474	0.0139
	LBA	15	7	4		1.1	0.3	0.981 ± 0.652	0.0302	0.0170
	vLBA	15	7	5	$\lambda = 0.01$	1.0	0.0	0.929 ± 0.585	0.0310	0.0147
	rvLBA	15	7	5	$\lambda = 0.03$, $\epsilon = 0.1$	1.0	0.0	0.958 ± 0.628	0.0298	0.0158
	sLBA	15	7	4	'tmean'	1.1	0.3	0.954 ± 0.632	0.0292	0.0168
1.0	svLBA	15	7	5	$\lambda = 0.005$, 'wmean'	1.1	0.3	0.920 ± 0.579	0.0292	0.0152
	STA	11	7	2	$K_\sigma = 4.9$	0.0	0.3	1.382 ± 2.196	0.4583	0.4056
	AMA	15	7	6	$K_\sigma = 7.0$	1.0	0.4	1.445 ± 1.730	0.0536	0.0270
	HSA				$\lambda = 0.5$	0.7	0.4	1.849 ± 1.422	0.0550	0.0318
	PIV	32	3			1.0	0.0	1.434 ± 2.667	0.0573	0.0187

more than two parameters and affine motion models. However, the affine motion models were less suitable to accurately model the flow field in small neighborhoods than the learned motion models applied by the LBA. Therefore, the learned motion models were superior for this sequence. The worst results were obtained with the STA and especially with the HSA. The performance of the PIV approach was good for NSR = 0, but for increasing noise rations the AAE increased relatively fast.

Apart from the AAE also the ADE_x and the ADE_y are given in Table 5.1. Generally the findings of the different error measures match each other with exception of the STA. Here, the relatively large displacement errors are due to sporadic outliers with very large displacements. These outliers have a strong influence on the ADE. The effect on the AAE is rather little. Nevertheless, it leads to an increased standard deviation. In order to improve the results obtained with the STA, a sparsification of the flow field on the basis of a confidence measure (cf. Section 3.4.4) may be an option.

So far, the motion models were learned from sample patches that were rotated, mirrored and time reflected in order to remove any directional bias from the models as shown in Section 4.1.2. However, one may argue that in some cases, e.g., if the training set is taken from the correct

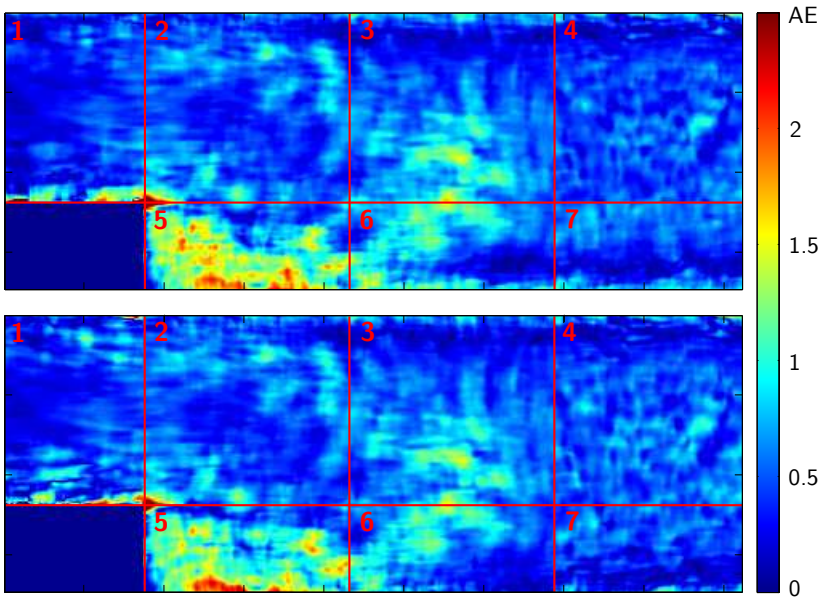

Figure 5.10: AE of the BFS sequence derived with the lLBA. The image is divided into seven areas, ignoring the step region. *Top*: Results obtained with motion models learned with patch transformations. *Bottom*: Results obtained with motion models learned without patch transformations.

Table 5.2: AE of the different areas denoted by the red rectangles in Figure 5.10 obtained with or without patch transformations (pt). The lowest error value of each area is highlighted in blue.

	AAE [°]	
region	with pt	without pt
'1'	0.406 ± 0.292	0.447 ± 0.287
'2'	0.525 ± 0.230	0.528 ± 0.231
'3'	0.568 ± 0.256	0.563 ± 0.244
'4'	0.436 ± 0.166	0.480 ± 0.184
'5'	1.049 ± 0.486	0.983 ± 0.446
'6'	0.693 ± 0.316	0.624 ± 0.297
'7'	0.471 ± 0.202	0.413 ± 0.193
'full'	0.541 ± 0.348	0.537 ± 0.347

flow field or is at least very similar to it, it might be better to retain the directional biases in the models as they reflect actual biases that exist in the data. It might further be of interest, to what extent a restriction of the training data to local areas of interest would affect the results. This restriction of the training data would generate motion models, which are specific for certain image regions. One interesting region in the BFS sequence is for instance the recirculation area, which contains different flow features than a region further downstream.

The AAE obtained with the lLBA and the first five motion models of size $15 \times 15 \times 7$ is shown in Figure 5.10. The upper image shows the AE that was obtained by using motion models learned with patch transformations, whereas the lower image shows the errors obtained by using motion models learned without patch transformations. The different regions marked by the red lines indicate the areas from which different sets of motion models were learned, once with and once without patch transformations. In total, the images were divided into seven regions, whereas the step region, which contains no fluid flow, was ignored. The estimation of the flow field was then restricted to these regions and the motion models applied for each region were

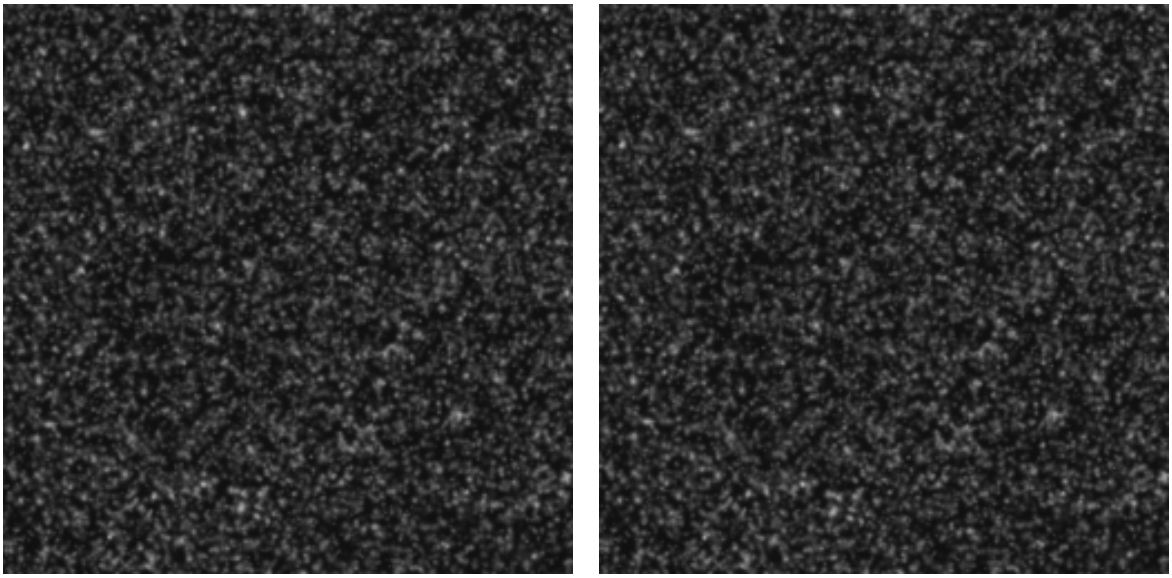

Figure 5.11: Two subsequent frames of the 2D turbulent image sequence.

the ones learned from the same region. In Table 5.2 the AAE of each region, labeled by the red numbers, is shown for both sets of motion models with and without patch transformations. Prior to the estimation of the flow field the images were smoothed with a Gaussian kernel of standard deviation $\sigma = 1.5$.

Figure 5.10 and Table 5.2 show that the largest errors occur in the recirculation area ('5'). This is also the area with one of the greatest differences of the AAE between using motion models learned with patch transformations and using motion models learned without patch transformations. The application of motion models learned without patch transformations led to a reduction of the AAE of approximately $0.07\,^\circ$ which corresponds to $6\,\%$ in this region. In region '6' and '7' a similar improvement could be obtained by not using patch transformations. However, in other areas especially area '1' and '4' the application of patch transformations yielded lower error values. Therefore, the overall improvement achieved by not using patch transformations was only marginal.

Since the application of patch transformations yields more generalized motion models, with a wider field of application, these transformations are reasonable. However, if the field of interest lies within a certain flow region, motion models specially trained for this region may be a good alternative and, therefore, patch transformation might be counterproductive in this case. Generally, the additional effort of constructing special motion models for every image region is not proportional to the benefit.

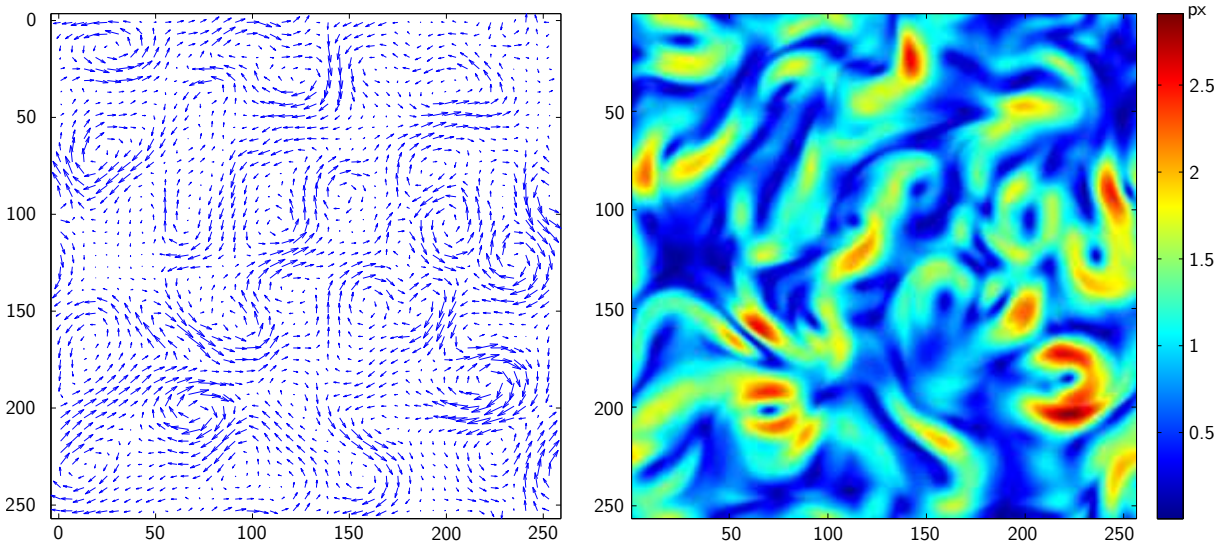

Figure 5.12: 2D turbulent sequence. *Left*: Vector plot of the correct flow field. *Right*: Color coded magnitude of the displacements.

5.2.2 2D turbulence

The sequence

This test case consists of a synthetic image sequence of a self sustained 2D turbulent flow provided by Carlier [2005a]. The velocity field was computed by solving the vorticity transport equation in Fourier space, and a highly non-rigid sequence with self sustained 2D turbulence was constructed. Large scale motions of the atmosphere or the ocean can be modeled, due to their thinness, in a simplified way with the 2D Navier-Stokes equation and are, therefore, examples of 2D turbulent flows [Ferziger et al., 2002].

The images are of size 256×256 px^2. In Figure 5.11 two subsequent frames of the sequence are shown. The maximum displacement is approximately 3 px, whereas the mean displacement is about 1 px. In order to study the ability to handle noisy conditions, Gaussian noise was added to the images as it was done for the BFS image sequence described in Section 5.2.1. Again the NSR was enlarged in steps of 5 % from 0 % to 100 %. A vector field of the sequence is depicted in Figure 5.12 together with the color coded magnitude of the displacement vectors on the right side.

Results

Once again, the influence of the used training data on the resulting flow fields was estimated. Accordingly, the motion models shown in Figure 5.3 with exception of 'BFS-PIV' were used. Representative for all versions of the LBA, the influence of the applied motion models was investigated with the lLBA. For low noise ratios, the preferred dimension of the motion models was $11 \times 11 \times 3$, whereas for larger noise ratios an increased temporal dimension with a total

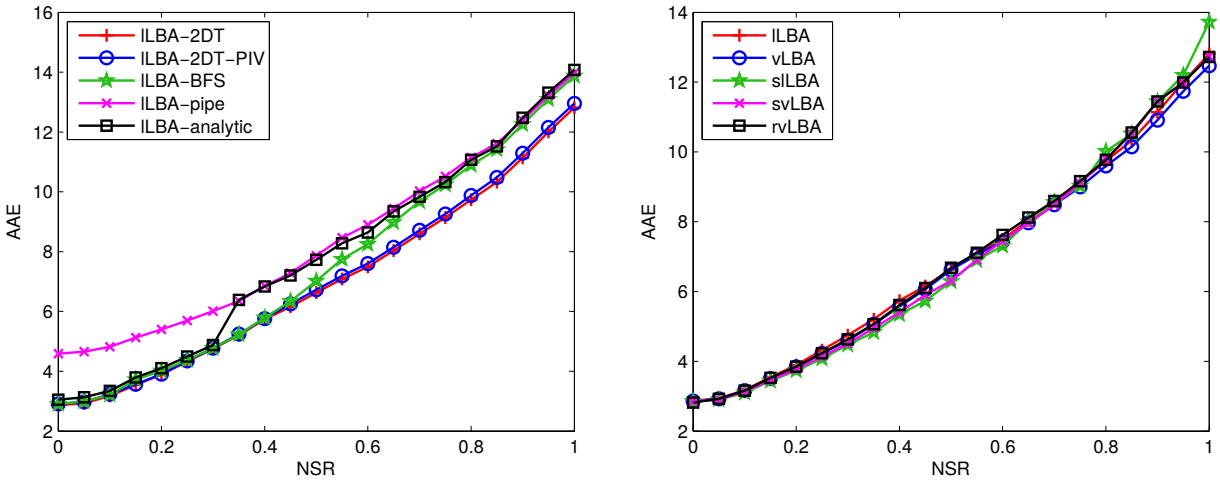

Figure 5.13: AAE of the LBA in dependance of NSR. *Left*: Results obtained with the lLBA with different training data. *Right*: Results obtained with different versions of the LBA.

motion model size of $11 \times 11 \times 7$ yielded better results. Also the ideal number of motion models was different for different NSR. Depending on the applied motion models, the ideal number ranged for low noise ratios from $K \approx 20$ for the models '2DT', '2DT-PIV', and 'BFS' to $K \approx 10$ for the 'pipe' and 'analytic' models. For higher noise levels, the ideal number ranged between $K = 5$ and $K = 3$. The images were preprocessed by smoothing with a Gaussian kernel in spatial and in temporal direction. With increasing noise ratio also the standard deviations of the Gaussian kernels in spatial (σ_ρ) and in temporal (σ_t) direction were increased.

The results in form of the AAE achieved with the lLBA are shown in the left plot of Figure 5.13. The lowest AAE was obtained with the motion models learned from the correct flow fields ('2DT') as well as the motion models learned from the PIV flow fields ('2DT-PIV'). For all noise levels, the difference between these two cases was only marginal. Comparable results were also obtained with the 'analytic' and the 'BFS' motion models for NSR < 0.3 and NSR < 0.5, respectively. However, for larger noise levels the AAE obtained with these two sets of motion models was approximately $1.5\,^\circ$ higher than the AAE achieved with '2DT' or '2DT-PIV'. Permanent higher error values were obtained with the motion models learned from the laminar pipe flow ('pipe'). As can be seen in Figure 5.3 the 'pipe' set of motion models does not include a rotation mode, which is essential to model the turbulent flow of this sequence. As already described in Section 4.1.2, the relevant flow features of the sequence of interest must be included in the training data. Otherwise, these flow features are completely missing in the set of learned motion models and cannot be modeled correctly. Therefore, the motion models learned from the laminar pipe flow are not suitable for the 2D turbulent sequence. Since the motion models learned from the BFS ('BFS') and from the analytic set ('analytic') contain the relevant flow structures, these models are at least for low noise levels capable to estimate the flow field. The influence of the rotation mode was also explored estimating the flow field with the '2DT' models excluding the rotation mode. Omitting this mode led to an increase of the AAE from $2.86\,^\circ \pm 2.12\,^\circ$ to $4.04\,^\circ \pm 3.70\,^\circ$.

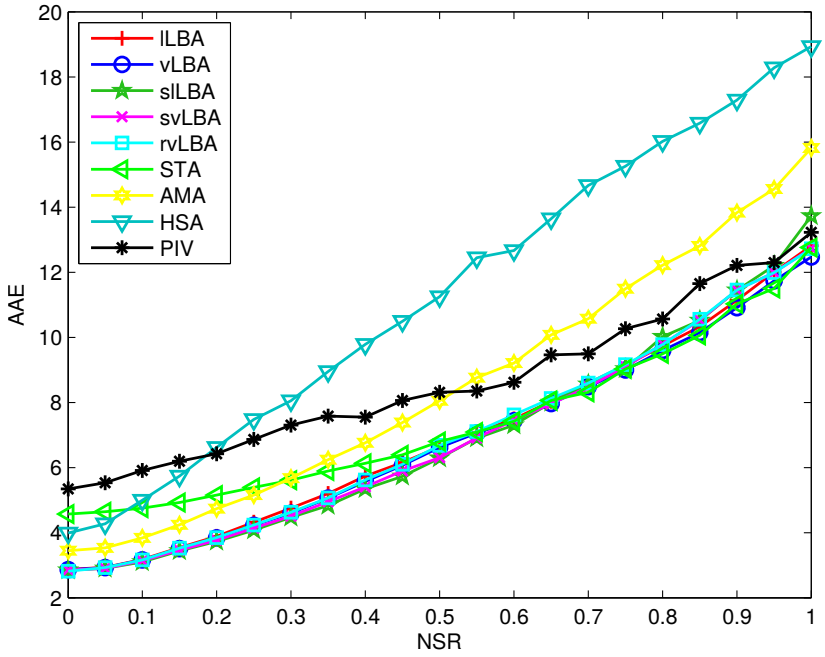

Figure 5.14: Comparison of different optical flow techniques on the 2D turbulent sequence, including the LBA, the STA, the AMA, the HSA as well as PIV.

In the following the '2DT' motion models were used.

The results obtained with different versions of the LBA are compared in the right plot of Figure 5.13. The error values of all methods are very similar. However, there are small differences, but unlike the BFS sequence (cf. Figure 5.7) there is not one single version of the LBA that is better than the other approaches for all noise levels. The vLBA was slightly better than the lLBA. The statistical approaches were best for low error levels but for high error levels they performed worse than the non-statistical approaches. The results obtained with the rvLBA were neither the best nor the worst.

In Figure 5.14 the results are compared to the ones achieved with other methods. The same optical flow and PIV approaches were used than described for the BFS sequence in Section 5.2.1. For each method and each noise level, the settings were chosen for which the lowest AAE could be obtained. The error values as well as the chosen parameters are given in Table 5.3 for three different noise levels. Additionally to the AAE also the ADE_x and ADE_y are listed. As previously described for the BFS sequence, the relatively large ADE of the STA compared to the AAE shown in Table 5.3 is due to sporadic wrong displacement estimates, that are extremely large. These outliers have a strong influence on the ADE.

Similar to the BFS sequence, the LBA was superior. It performed better than all other tested methods. However, the STA obtained similar results than the LBA but merely, for NSR > 0.5. For lower NSR, the AAE obtained with the STA was up to 60 % higher than the AAE obtained with the LBA. If the results of the STA and the AMA are compared, it is conspicuous that for low noise levels the AAE of the AMA is smaller, whereas for high noise levels it is the other way round and the AAE of the STA is smaller. This fits to the findings that the ideal number of used motion models decreases with increasing noise ratios (cf. Table 5.3). This behavior indicates

Table 5.3: Performance of different methods on the 2D turbulent sequence. All parameters have been optimized for each method to yield the lowest possible error measures. Highlighted in blue is the lowest AAE of each noise level.

NSR	method	ρ	t	K	other parameters	σ_ρ	σ_t	AAE [°]	ADE$_x$ [px]	ADE$_y$ [px]
	LBA	11	3	21		0.0	0.0	2.862 ± 2.122	0.0723	0.0662
	vLBA	11	3	21	$\lambda = 0.0001$	0.0	0.0	2.816 ± 2.083	0.0716	0.0656
	rvLBA	11	3	21	$\lambda = 0.0001$, $\epsilon = 10$	0.0	0.0	2.815 ± 2.081	0.0717	0.0655
	sLBA	11	7	22	'wmean'	0.0	0.0	2.848 ± 2.074	0.0706	0.0669
0.0	svLBA	11	7	19	λ=0.0001, 'wmean'	0.0	0.0	2.852 ± 2.086	0.0715	0.0677
	STA	11	7	2	$K_\sigma = 1.0$	0.0	0.0	4.573 ± 3.999	0.1225	0.1063
	AMA	11	3	6	$K_\sigma = 1.5$	0.0	0.0	3.314 ± 2.496	0.0869	0.0793
	HSA				$\lambda = 0.035$	0.0	0.0	3.989 ± 2.943	0.1104	0.1030
	PIV	8	1			0.3	0.0	5.346 ± 3.935	0.1072	0.1324
	LBA	11	7	7		1.4	0.4	6.631 ± 4.198	0.1776	0.1636
	vLBA	11	7	7	$\lambda = 0.00005$	1.4	0.0	6.618 ± 4.157	0.1764	0.1627
	rvLBA	11	7	7	λ=0.00007, $\epsilon = 2.0$	1.4	0.0	6.681 ± 4.293	0.1755	0.1622
	sLBA	11	7	7	'mean'	1.4	0.4	6.289 ± 3.870	0.1693	0.1528
0.5	svLBA	11	7	7	λ=0.00005, 'mean'	1.4	0.0	6.307 ± 3.843	0.1710	0.1543
	STA	11	7	2	$K_\sigma = 2.5$	0.0	0.6	6.797 ± 5.166	0.2558	0.1699
	AMA	11	7	6	$K_\sigma = 4.0$	1.4	0.4	8.049 ± 4.721	0.2204	0.1971
	HSA				$\lambda = 0.07$	1.1	0.4	11.252 ± 6.202	0.2829	0.2660
	PIV	16	3			0.6	0.0	8.310 ± 6.053	0.1783	0.1915
	LBA	11	7	5		2.0	0.4	12.813 ± 7.568	0.3409	0.2919
	vLBA	11	7	5	$\lambda = 0.00024$	2.1	0.0	12.467 ± 6.903	0.3328	0.2809
	rvLBA	11	7	5	λ=0.00027, $\epsilon = 2.5$	2.1	0.0	12.724 ± 7.567	0.3267	0.2774
	sLBA	11	7	5	'med'	2.0	0.4	13.725 ± 8.880	0.3400	0.2922
1.0	svLBA	11	7	5	λ=0.00025, 'tmean'	2.0	0.0	12.738 ± 7.358	0.3307	0.2791
	STA	11	7	2	$K_\sigma = 3.5$	0.0	0.4	12.762 ± 11.975	1.9089	3.3895
	AMA	15	7	6	$K_\sigma = 5.0$	1.6	0.4	15.816 ± 9.206	0.3966	0.3588
	HSA				$\lambda = 0.095$	1.5	0.4	18.936 ± 9.987	0.4583	0.4028
	PIV	32	3			1.0	0.0	13.224 ± 9.388	0.3030	0.2945

that the complexity of the motion models must be decreased for higher NSR. If small scale structures are included in the motion models, they try to model the noise of the image sequence as small scale flow structures. Therefore, small scale motion models must be omitted for noisy images. Apart from small noise levels, the results of the HSA are the worst. For small NSR, the highest AAE values were obtained with PIV. However, with increasing NSR the performance of PIV becomes better, and for very high noise ratios the error values are similar to the error values obtained with the LBA.

In Figure 5.15 the energy spectrum (cf. Section 2.1.3) of the horizontal velocity component is compared for the lLBA, the vLBA, and PIV. The flow fields were estimated from the noise free image sequence with the parameters given in Table 5.3. Additionally, the energy spectrum of the correct ground truth flow field (gt) is shown. The energy spectrum is often used to describe the characteristics of turbulent fluctuations. Normally a decrease of the spectrum proportional to k^{-3} would be expected for 2D turbulence [Boffetta, 2007]. However, in the energy spectrum shown in Figure 5.15 a decrease proportional to k^{-6} is observed. Due to the logarithmic scale, this corresponds to a slope of -6. This discrepancy could be due to the nature of this simulated

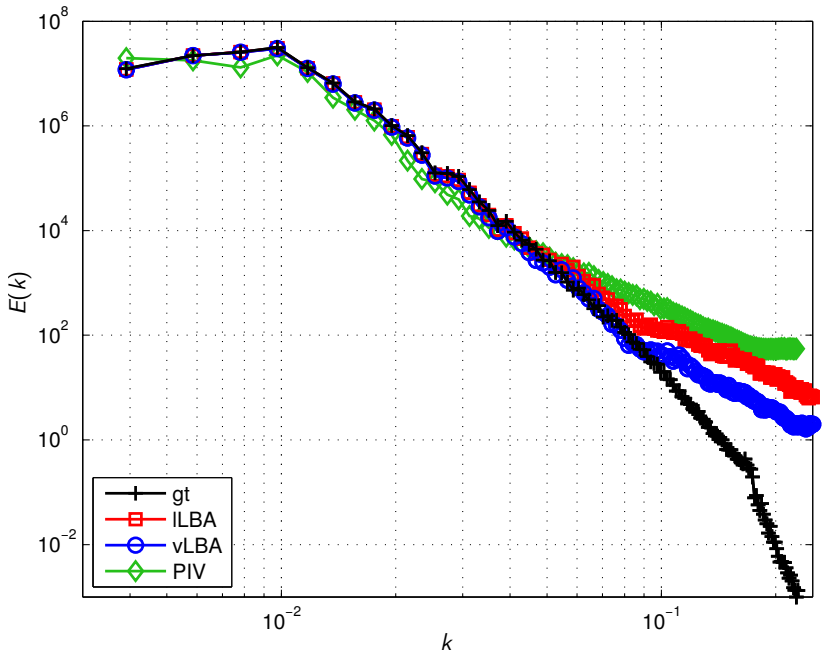

Figure 5.15: Energy spectrum of the horizontal velocity component shown for ground truth (gt), lLBA, vLBA, and PIV.

self sustained 2D turbulence, since the slope shown in Figure 5.15 nicely matches the slope found by Heitz et al. [2010] for the same sequence.

For large scales (small k), the two LBA spectra are very similar to the ground truth spectrum. This indicates a correct estimation of the large scale motions by the LBA. The PIV spectrum on the other side slightly differs from the correct spectrum. For small scales (large k), all estimated spectra show a different trend than the ground truth spectrum. Overall the best agreement with the ground truth spectrum is found for the vLBA. This behavior is also reflected by the error values given in Table 5.3. The table indicates that the AAE obtained with PIV is approximately 90 % higher compared to the AAE obtained with the vLBA.

The LBA also provides additional information about local flow structures by means of the

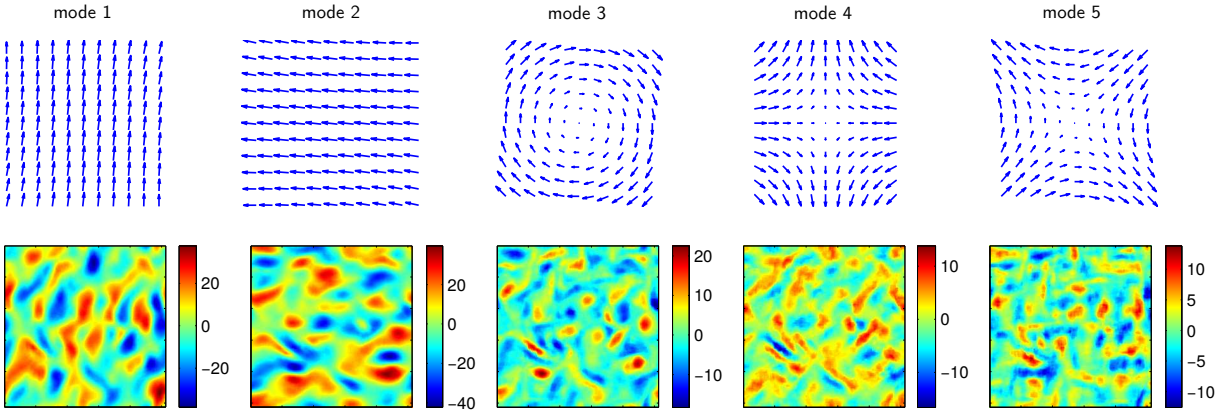

Figure 5.16: From *left* to *right* the first five motion models of size $11 \times 11 \times 1$ (*top row*) are shown together with the parameter images (*bottom row*).

learned motion models and the coefficients of the linear combination given in Equation (4.1). The estimated flow field is composed of typical flow structures embodied by the learned motion models and the coefficients, which indicate the weight of the flow structures at each location. In Figure 5.16 the first five motion models are shown in combination with the parameter images, which indicate the weight of each motion model for all image locations. Since the learned motion models are in fact the POD modes of the training data, this decomposition is comparable to a POD of the estimated flow field itself. At least if the training data and the observed flow fields are similar. Therefore, these motion models can be used to identify dominant structures of the flow field (cf. Section 2.2). Of course, this is only possible for structures, which are of similar dimension than the motion models. For the identification of large scale structures, the POD would have needed to be performed on the entire flow field instead of only on small local patches.

A striking structure shown in Figure 5.16 is for instance represented by the third motion model, which contains pure rotation. The corresponding parameter image indicates areas where rotations are present. In order to verify this statement, the vorticity as well as the *Q-criterion* of Hunt et al. [1988] were computed. In 2D, the vorticity is defined by

$$\nabla \times \boldsymbol{u} := \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \tag{5.1}$$

and indicates vortex regions by values larger than a threshold defined by the user. However, the vorticity cannot distinguish between swirling motion and shearing motion [Kida and Miura, 1998]. Therefore, often other methods are used in order to identify vortex regions such as for instance the Q-criterion. It defines a vortex as a spatial region, where the Euclidean norm of the vorticity tensor $\Omega = \frac{1}{2}\left(\nabla \boldsymbol{u} - (\nabla \boldsymbol{u})^T\right)$ dominates that of the rate-of-strain tensor $S = \frac{1}{2}\left(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T\right)$. The Q-criterion is defined by

$$Q := \frac{|\Omega|^2 - |S|^2}{2} > 0\,. \tag{5.2}$$

In the case of 2D flows, the same criterion has been known as the elliptic version of the Okubo-Weiss criterion [Haller, 2005]. The parameter image of the third rotation mode, the vorticity, and the Q-criterion image are shown together with the estimated velocity field in Figure 5.17. Visually, the parameter image and the vorticity image are very similar to each other and predict the same structures. Some of the vortices, which could be identified in all three images, are marked by the black circles in Figure 5.17. In essence, all of the vortices identified by the Q-criterion could also be identified by the parameter and the vorticity image. However, there are also some structures such as the one marked by the red rectangle, which were only indicated in the two later images, and might be due to shearing motion.

Nevertheless, the decomposition of the flow field in the typical motion models weighted by the parameters provides useful information about the underlying flow field. The motion models,

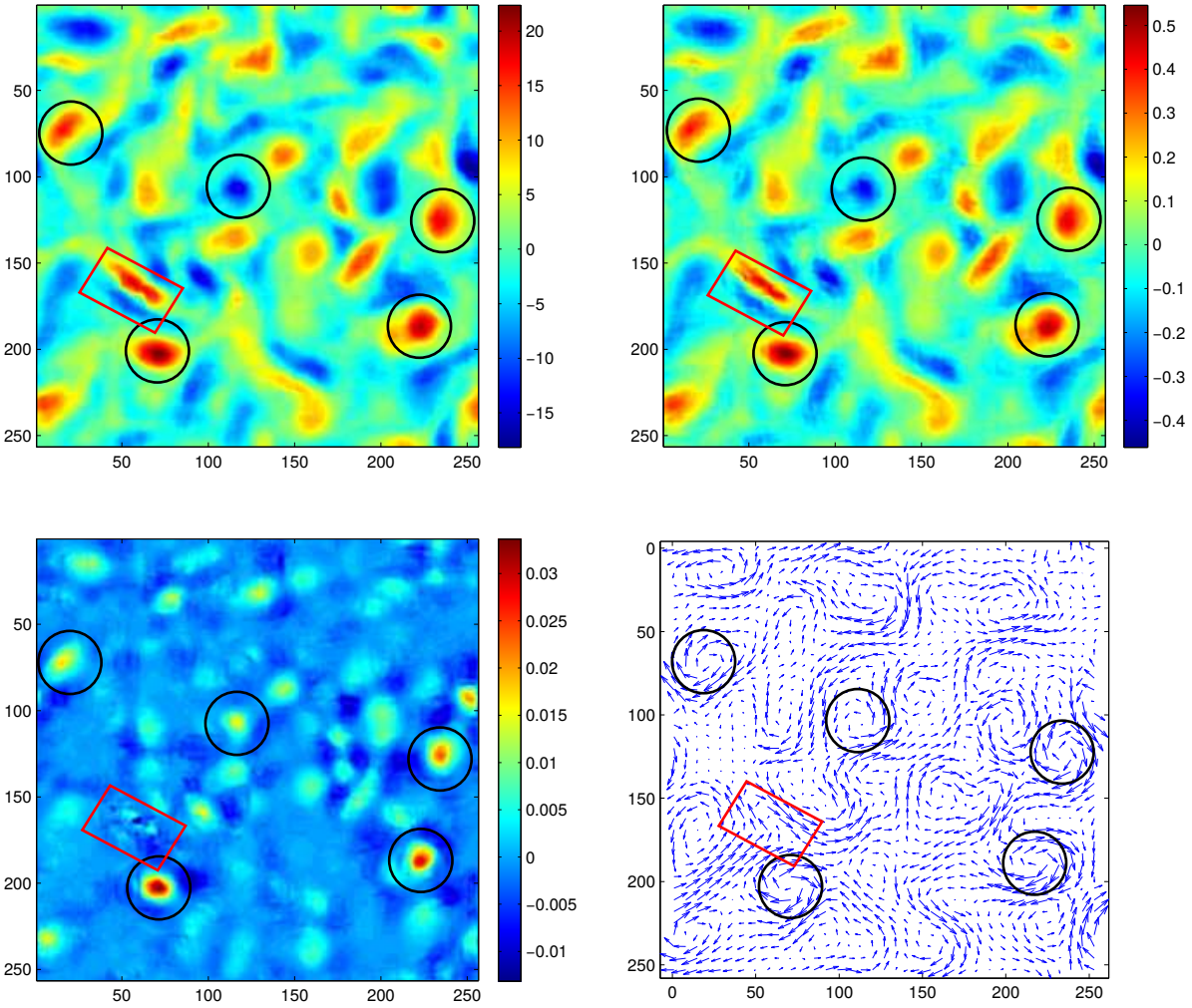

Figure 5.17: Comparison of the parameter image of the third motion model (*top left*) with the computed vorticity (*top right*) and the Q-criterion image (*bottom left*). Additionally the vector plot of the estimated flow field is shown (*bottom right*). Some of the identified curls are marked by black circles.

which are at hand anyway for the LBA, describe the most significant motions and dominant structures that are present in the flow.

5.2.3 Real backward facing step

In order to test the ability of the LBA to determine the displacement field of real world particle images, the turbulent flow over a BFS was employed. The image sequence was recorded by Weier et al. [2011]. In Figure 5.18 two subsequent frames of the sequence are depicted. Unlike the synthetic BFS sequence, the inflow region and the step itself is not visible in the images of the real BFS sequence. Within the field of view is the area directly located behind the step. The flow direction was like in the synthetic case from left to right. The PND is about 0.05 ppp and the step height Reynolds number of the flow was 1875. The maximum in-plane displacement between subsequent images is approximately $7 - 8$ pixels. Due to this relatively large displacements

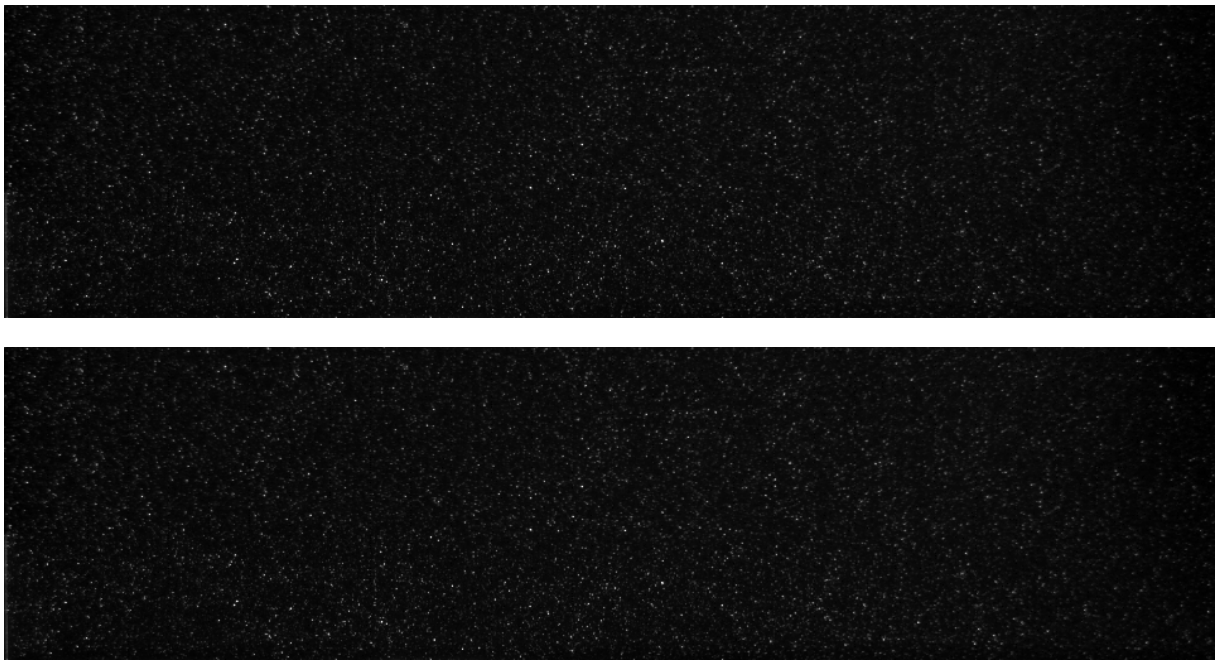

Figure 5.18: Two subsequent frames of the real BFS image sequence.

the LBA had to be applied together with a coarse-to-fine strategy. Therefore, the hierarchical multi-scale approach introduced in Section 3.4.6 was used.

As the sequence consists of real images, the correct displacement field is unknown. Therefore, the AAE and the ADE cannot be calculated. In order to validate the results quantitatively, the AIE (cf. Section 3.5.3), which is defined as the RMS of the gray value differences of the real and the warped version of corresponding images, was used. However, this error measure is only a rough indicator for the quality of the estimated flow field, since it merely compares images and not the flow field itself. This is especially a problem in homogeneous image regions, since here wrong flow estimates may lead to low error values.

The flow field was estimated with the LBA and compared to the results obtained with PIV. In Figure 5.19 two flow fields obtained with the svLBA and PIV are compared visually. The flow fields shown in the figure are mean flow fields averaged over ten subsequent flow fields. Both flow fields look very similar and by eye almost no differences are visible. Only in the curl area and in the bottom right corner the images look slightly different. In order to learn the motion models used for the svLBA, the simulated flow fields of the synthetic BFS sequence (Section 5.2.1) as well as flow fields obtained with PIV were applied as training data. Virtually, there were no differences detected between the resulting flow fields, and the AIE was the same in both cases. The size of the motion models was $15 \times 15 \times 1$. Due to the applied multi-scale approach, the application of a larger temporal dimension was not possible. The best results were obtained with $K = 5$ motion models and a regularization parameter $\lambda = 0.0021$. Prior to the estimation of the flow field the images were smoothed with a Gaussian kernel K_σ of

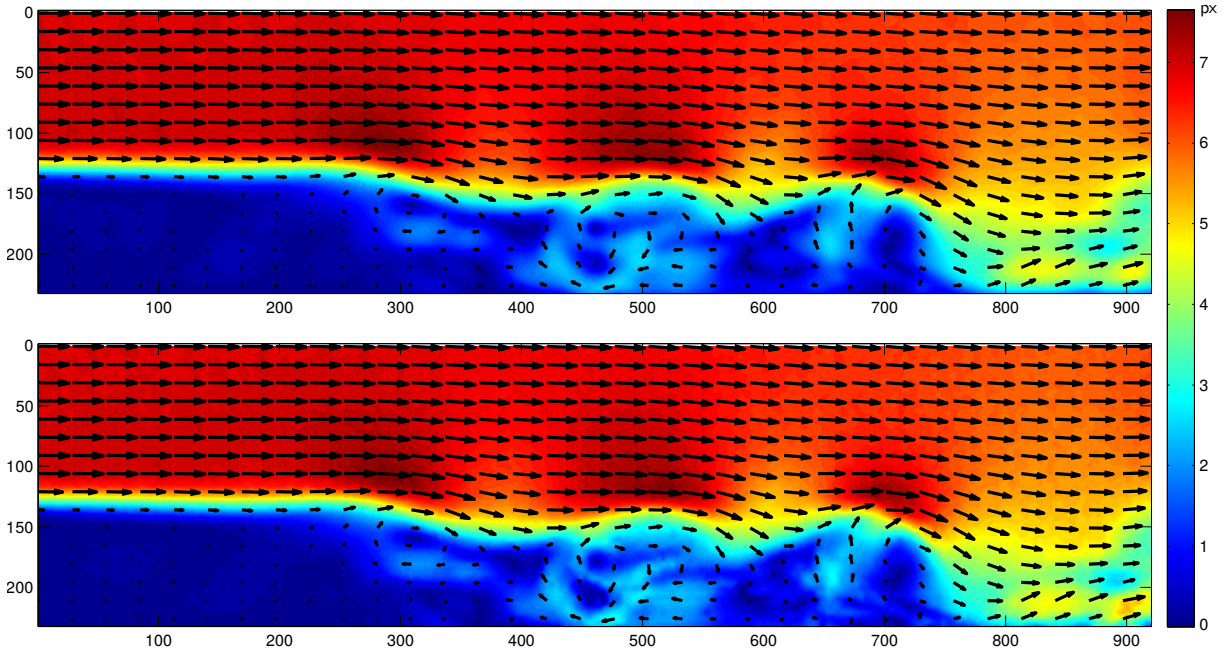

Figure 5.19: Mean flow fields of the real BFS sequence estimated with the svLBA (*top*) and PIV (*bottom*). The magnitude of the displacement vectors is displayed color coded.

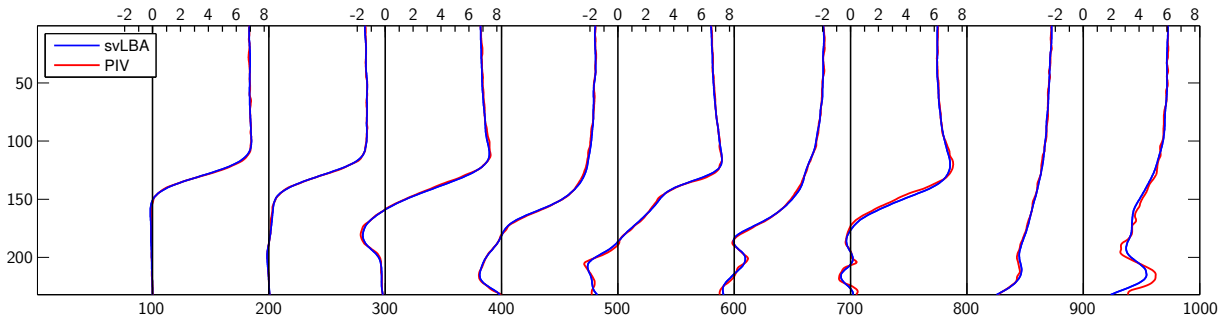

Figure 5.20: Horizontal velocity profiles of the mean displacement fields at different positions.

standard deviation $\sigma = 0.4$. As averaging method the weighted arithmetic mean was used and the statistical sample consisted of 121 ($N_\rho = 11$) vector patches. The achieved AIE with the svLBA was 0.0242. The results obtained with other versions of the LBA were very similar and the AIE was only marginally larger. PIV was performed with fluere 1.3 (cf. Section 3.3.3) with an initial interrogation window size of 64×64 and a final interrogation window size of 16×16. The AIE obtained with PIV was 0.0246. This value is only about 1.5 % higher than the value obtained with svLBA. Thus, the differences of the two approaches are quite small.

This can also be seen in Figure 5.20 where horizontal velocity profiles of both methods are shown at regularly spaced positions with a distance of 100 pixels. A displacement scale is shown above every second profile and given in terms of pixels. The profiles were determined from the mean displacement fields shown in Figure 5.19. Especially the profiles located at the left side are almost identical for the svLBA and PIV. The red profiles of PIV are hardly visible behind the

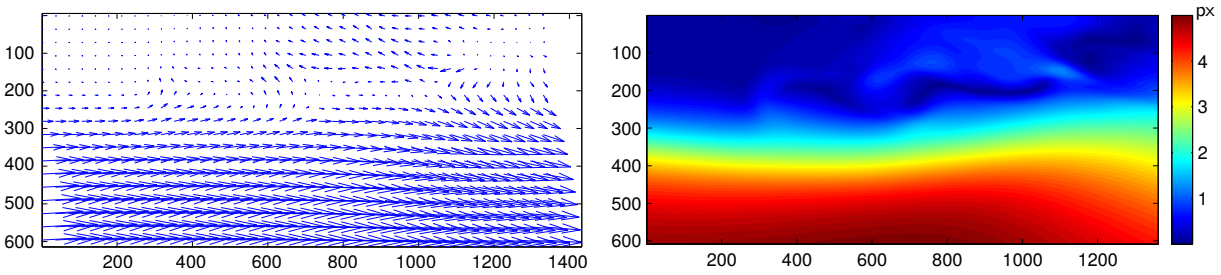

Figure 5.21: Displacement vector field (*left*) and magnitude (*right*) of the laminar separation bubble sequence.

blue ones of the svLBA. However, since the profiles of the svLBA are a bit smoother, the wilder PIV profiles sometimes emerge behind them. Further down stream, at the right side of the plot, small differences between single profile pairs occur. The largest discrepancy of approximately 0.5 px can be found on the last profile pair.

This example shows, that an estimation of the displacement field from real particle images with the LBA is feasible. Despite the relatively large displacements, which are in fact ideal for PIV, a meaningful displacement field could be obtained, which is comparable to the one obtained with the well established PIV method.

5.2.4 Laminar separation bubble

In order to test the LBA on a competitive PIV image sequence, for which correct flow fields are available, a test case from the international PIV Challenge [PIV-Challenge, 2014] was chosen. So far the PIV Challenge was held four times namely in the years 2001, 2003, 2005 and 2014. The objective was to exchange the knowledge of PIV image analysis techniques and to assess the current state of the art of PIV as well as to guide future development efforts. At each edition several challenging test sequences were provided and the contributing teams proposed different algorithms in order to estimate the flow fields. In this way a fair comparison of different techniques was possible, which led to the construction of improved estimation techniques.

As test case a sequence from the third international PIV Challenge [Stanislas et al., 2008] was taken, since the sequences of the latest PIV Challenge had not yet been available. Selected was case B, which consists of a direct numerical simulation of a laminar separation bubble. The images of the synthetic sequence are of size 1440×688 px^2. They were generated with equidistant time steps between subsequent frames. The PND is approximately 0.025 ppp and the maximal frame to frame displacement is around 5 px, whereas the mean displacement is around 2.2 px. In Figure 5.21 the velocity field is depicted together with the color coded magnitude. A special challenge of this sequence is the large dynamic range as well as strong gradients. The smallest displacements occur inside the laminar separation bubble located at $x \in [0, 600]$, $y \in [0, 200]$, whereas the largest displacements occur near the bottom of the image. The flow field can be divided into an upper half with small displacements and a lower half with relatively large displacements. In order to simulate different light sheet intensities as well as different particle

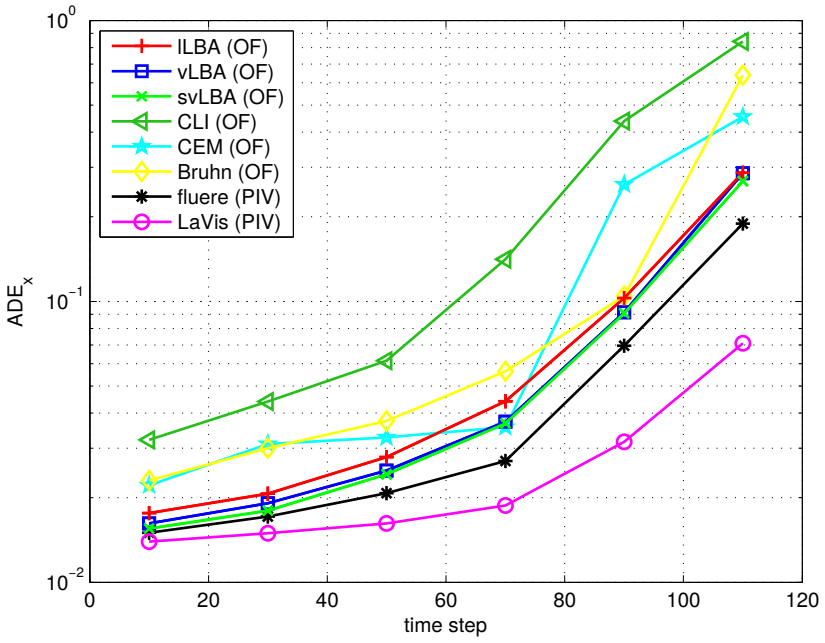

Figure 5.22: Performance of different methods on the laminar separation bubble sequence. With increasing time the signal-to-noise ratio decreases leading to higher noise levels at later time points.

diameters and different sensor sensitivities, the signal-to-noise ratio was decreased with time [Stanislas et al., 2008].

The displacement field of the sequence was estimated with the LBA and compared to the ones of other estimation methods. Because of the relatively large displacements, again a coarse-to-fine strategy was used, as it was done for the real BFS sequence in Section 5.2.3. The sequence was constructed to work best for interrogation window sizes of 32×32 and the optimal size of the motion models was $35 \times 35 \times 1$ and, thus, of similar dimension. The optimal number of motion models was $K = 5$ for low noise levels and decreased to $K = 2$ for high noise levels. As training data for the motion models, the flow fields obtained with PIV were used.

The LBA was compared to some of the approaches submitted to the PIV Challenge, namely the two optical flow approaches CEMAGREF/INRIA (CEM) and CLIPS/LIMSI (CLI) and the best correlation-based PIV approaches submitted by LaVision (LaVis), which is commercially available [LaVision, 2014]. The results of these methods were originally published in Stanislas et al. [2008]. The CEM approach is a variational optical flow approach based on the method proposed by Corpetti et al. [2002]. It is embedded in a multi-resolution scheme in order to handle large displacements. The CLI approach was developed by Quénot et al. [1998] and uses an hierarchical processing scheme to cope with large displacements. Another state of the art optical flow approach, which was not submitted to the PIV Challenge, but applied here, was the combined local and global method (cf. Section 3.4.4) proposed by Bruhn et al. [2005] (Bruhn). For the estimation of the flow field, the version implemented by Liu [2009] was used. This method also uses a coarse-to-fine strategy in order to handle large displacements. Apart from the PIV approach proposed by LaVision, also the PIV method applied in the previous sections (fluere 1.3) was used.

The results in form of the ADE in x-direction (ADE_x) are shown for different time steps in Figure 5.22. With time also the noise ratio increases. The plot is similar to the ones published in Stanislas et al. [2008] but has a logarithmic scale for a better representation of the data. It can be seen that the two correlation-based PIV methods are superior to all optical flow methods. This is not surprising, since the sequence was specially designed to meet the requirements of PIV methods (e.g., large displacements and small particle sizes). However, among all compared optical flow methods the LBA obtained the best results. The plot shows the results of the lLBA, the vLBA, and the svLBA. Out of these approaches the svLBA performed best. At the lowest noise ratio the results are only 3.5 % worse than the results obtained with fluere 1.3 and 10 % worse than the results obtained with LaVision. However, for larger noise ratios the discrepancy is larger. Nevertheless, for low noise ratios the error obtained with the svLBA is approximately 40 % lower than the error obtained with the next best optical flow approach. These findings show that the LBA could not quite keep up with the best PIV methods, but it outperformed other state of the art optical flow approaches, which were also designed for the determination of fluid flows from particle images.

5.3 Conclusion

This chapter describes the ideal properties and conditions, which have to hold for the used particle images, as well as the performance of different approaches on four test sequences. For the LBA, a PND larger than 0.1 ppp and particle displacements smaller than 1-2 pixel are ideal. This is in contrast to the optimal conditions of PIV, which mainly prefers larger displacements. To handle large displacements, the LBA must be performed within an hierarchical multi-scale framework. However, for small displacements, the non-hierarchical LBA is more accurate.

The performance of the different versions of the LBA on four test sequences was compared to the performance of common optical flow methods as well as PIV. This led to the following findings. On the synthetic image sequences of a BFS and a 2D turbulence, the LBA obtained the lowest error values for all noise levels. Especially the svLBA performed very well. With increasing noise level, it is advisable to reduce the complexity of the flow model by using lesser motion models. The application of motion models learned from flow fields, which were previously obtained with PIV, led to improved flow fields compared to the PIV flow fields themselves. Therefore, the LBA can also be understood as post-processing step leading to improved accuracy. The motion models that are at hand anyway for the LBA, provide together with the parameter images additional information about dominant flow events and coherent structures, which can be used to characterize the flow. The LBA also worked well on real images such as the real BFS sequence as well as on a more competitive sequence taken from the third international PIV Challenge. On the latter sequence, the LBA yielded results of improved accuracy compared to all other optical flow methods. For low noise levels, the ADE was hardly higher than the ADE obtained with PIV. Considering all results, it shows that for relatively small displacements,

e.g., one pixel, the LBA was able to outperform all other tested methods including PIV. If it is known beforehand that the LBA will be applied, measurements should be conducted with an appropriate frame rate in order to yield small particle displacements.

6 Conclusion and Outlook

6.1 Conclusion

The scope of this thesis was the development of new approaches for a better estimation of dense fluid dynamical motion fields from particle images. The intention was to use prior knowledge in form of typical motion models for the approximation of local flow structures of the underlying flow fields. On the one hand, such prior knowledge is not available in common optical flow and PIV methods. On the other hand, optical flow methods using prior knowledge in from of physics-based constraint equations are extremely complex and very costly. Furthermore, they require the complete knowledge of the boundary conditions, which is usually not given. With the proposed *learning-based approach* (LBA), it was possible to apply prior knowledge of local flow structures in a simple, yet, efficient way. Therefore, concepts were established in order to express the solution in terms of an optimal system of orthogonal basis functions. Such a system was learned from appropriate training vector fields by performing a POD on a set of small vector patches drawn from the training data. The training data consisted of known fluid dynamical velocity vector fields and by careful choice of the training data the motion models could be tuned for specific flow problems. The POD is a common tool in fluid dynamics and is often used to identify dominant flow events and coherent structures as well as to reconstruct missing information of incomplete flow fields. Therefore, it is perfectly suited for the determination of typical flow structures.

The approach was embedded into existing well-established frameworks known from computer vision. In essence, it was formulated as parametric optical flow approach, which was named *local learning-based approach* (lLBA). This local method was also extended to fit into a global context yielding the *variational learning-based approach* (vLBA) by using the concepts of variational and combined local global optical flow approaches. A further extension was derived by using non-quadratic error norms to obtain the *robust variational learning-based approach* (rvLBA). The robust error norm reduces the weights of outliers in order to lessen their influence on the computed solution. In all of these approaches, each flow vector was estimated in dependence of a small spatio-temporal neighborhood. Thereby, only the central flow estimate was used while the surrounding estimates were discarded. By using this available surrounding information, a large sample of estimates could be derived for the very same flow vector. The final flow estimate was then obtained from the data sample applying an averaging operation. This statistical extension was used in combination with the lLBA and the vLBA in order to obtain the *statistical local*

learning-based approach (slLBA) and the *statistical variational learning-based approach* (svLBA), respectively.

The properties and the performance of the different versions of the LBA were investigated on several synthetic flow sequences. In essence, the effect on the resulting flow field, of applying different numbers and kinds of spatio-temporal motion models of various sizes, was observed. The optimal number and the optimal size of the motion models depend on the complexity of the flow under study. Simple flows require only two to five relatively small motion models, whereas the ideal number and the ideal size of the motion models increase for flows of increased complexity, which contain many small scale structures. The inclusion of temporal information and, therefore, the application of spatio-temporal motion models led to improved results compared to the application of purely spatial motion models. However, experiments with noisy image sequences reveled that with increasing noise level the optimal number of motion models decreases. Considering too many motion models, the approach tries to model the occurring noise as small scale flow structures. Therefore, low energy motion models, which contain small scale structures must be omitted considering noisy sequences.

A comparison of the different versions of the LBA on the analytic sequences showed that the variational approach performed better than the local approach. The achieved AAE was in average 7 % lower. The statistical versions of the LBA were able to outperform the non-statistical versions. The error values of these methods were approximately 15 % and 8 % lower for the local and the variational version, receptively. With the rvLBA also a slight performance gain of about 3 % compared to the vLBA could be achieved. The improvements of the vLBA and the rvLBA compared to the lLBA are somewhat at the expense of the simplicity of the method. While for the purely local method a simple least squares approach is sufficient to solve the optical flow problem, the variational methods require more advanced techniques such as the minimization of a functional that consists of the data term and an additional regularization term. Especially the robust version requires extensive mathematical knowledge, since a non-linear problem must be solved. Yet, all mathematical concepts are similar to the concepts of common optical flow methods and are, therefore, well-known. Apart from improved results, the variational methods also yield dense flow fields due to the filling-in effect, which is not necessarily true for the local approach. The statistical approaches are only slightly more complex than the non-statistical versions. All of the information is readily available and an improved solution is obtained performing an averaging operation. Therefore, the statistical extension is a highly recommended add-on to the LBA.

The best conditions and image properties in order to apply the LBA are a relatively high PND (larger than 0.1 ppp) and particle displacements of about 1-2 pixels. If larger displacements are present in the sequence, the normal LBA leads to wrong estimates and a warping-based hierarchical multi-scale approach must be applied. However, together with the multi-scale approach only purely spatial and no spatio-temporal motion models can be used. Furthermore, the multi-scale approach cannot cope with small displacements. Therefore, it is recommended to use the non-

hierarchical version together with sequences of high frame rates and, thus, small displacements, whenever it is possible.

The LBA was also tested on different fluid dynamical applications in the presence of increasing noise. The results were compared to the results obtained with other common optical flow and PIV methods. Compared to the competing approaches, the LBA performed extremely well. Especially on the sequences with relatively small displacements such as the synthetic BFS and the 2D turbulent sequence the LBA obtained the lowest error values. On these two sequences, the AAE of the best LBA version was in average 25 % and 10 % lower than the AAE of the second best method. Tests on real images of a BFS showed that the LBA can also handle real world conditions. On a competitive sequence taken form the third international PIV Challenge, the LBA outperformed all other optical flow methods. The errors obtained with the LBA were only slightly higher than the errors obtained with PIV, even though, the image properties (e.g., relatively large displacements) were ideal for PIV. The LBA can also be regarded as post-processing step to improve the flow fields obtained with other methods such as PIV. Accordingly, the LBA can be performed using motion models learned from these existing flow fields in order to improve the results.

The LBA is a highly accurate method that is able to yield precise vector fields of fluid dynamical flows from particle images using prior knowledge about local flow structures. Yet, it is simple to implement and easy to apply. It is hardly more complex than common optical flow methods. The LBA also provides additional information about local dominant flow structures by means of the learned high energy motion models and corresponding parameters, which can be used to characterize the flow field.

6.2 Outlook

The proposed LBA could be improved in several ways. In order to cope with large displacements a combination of the LBA with correlation-based methods such as PIV is promising. Such an approach could link the advantages of both methods. PIV is a capable method for large displacements, whereas the LBA obtains dense flow fields and is highly accurate for small displacements. Accordingly, PIV could solve for a rough, large displacement field, which is afterwards refined by application of the LBA on a warped sequence, which merely contains small flow increments.

Due to the vast improvements obtained with the statistical extension of the LBA, further research in this direction seems reasonable. A thorough investigation of advanced statistical techniques connecting sample data from neighboring locations may uncover further improvements towards robuster estimates.

In order to reduce the computation time, the implementation of the approaches could be adapted. The local approach could be parallelized and implemented in graphics processing units (GPUs). The variational version could benefit from improved numerical multigrid schemes for a faster solution of the systems of equations. These improved implementations would be the first

step towards a real time performance of the LBA.

So far, only the case of 2D motion fields determined from particle images was considered. The LBA can also be applied in order to investigate 3D fluid flows. All equations can easily be extended to 3D. The assumption of constant image brightness is even more likely to hold in 3D, since if the entire measurement volume is homogeneously illuminated, there are no brightness variations due to out of plane motion, as it may be the case with 2D light sheets. In the last years a lot of progress was made in the development of volumetric velocimetry techniques such as tomographic PIV and the quantity and quality of real 3D sequences increased. In order to handle this data, accurate and reliable estimation methods are required.

Abbreviations

AE	*angular error*
AAE	*average angular error*
ADE	*average displacement error*
AIE	*average interpolation error*
AMA	*affine model approach*
BCCE	*brightness change constraint equation*
BFS	*backward facing step*
HSA	*Horn and Schunck approach*
LBA	*learning-based approach*
LIF	*laser induced fluorescence*
lLBA	*local learning-based approach*
NSR	*noise-to-signal ratio*
PIV	*particle image velocimetry*
PND	*particle number density*
POD	*proper orthogonal decomposition*
PTV	*particle tracking velocimetry*
RIC	*relative information content*
RMS	*root mean square*
rvLBA	*robust variational learning-based approach*
slLBA	*statistical local learning-based approach*
SOR	*successive over-relaxation*
STA	*structure tensor approach*
SVD	*singular value decomposition*
svLBA	*statistical variational learning-based approach*
TV	*total variation*
vLBA	*variational learning-based approach*

Bibliography

Edward H. Adelson and James R. Berger. Spatitemporal energy models for the perception of motion. *Journal of the Optical Society of America A*, 2:284–299, 1985.

Roland J. Adrian. Particle-Imaging Techniques for Experimental Fluid Mechanics. *Annual Review of Fluid Mechanics*, 23:261–304, 1991.

Roland J. Adrian and Jerry Westerweel. *Particle Image Velocimetry*. Cambridge University Press, 2011.

Ronald J. Adrian. Hairpin vortex organization in wall turbulence. *Physics of Fluids*, 19, 2007.

Padmanabhan Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2:283–310, 1989.

Alireza Bab-Hadiashar and David Suter. Robust Optic Flow Computation. *International Journal of Computer Vision*, 29(1):59–77, 1998.

Simon Baker, Daniel Scharstein, J.P. Lewis, Stefan Roth, Michael J. Black, and Richard Szeliski. A Database and Evaluation Methodology for Optical Flow. *International Journal of Computer Vision*, 92:1–31, 2011.

Wolfgang Bangerth, Ralf Hartmann, and Guido Kanschat. deal.II – a General Purpose Object Oriented Finite Element Library. *ACM Trans. Math. Softw.*, 33:24/1–24/27, 2007.

John. Barron, David Fleet, and Steven Beauchemin. Performance of Optical Flow Techniques. *IJCV*, 12:1:43–77, 1994.

Steven Beauchemin and John Barron. The Computation of Optical Flow. *ACM Computing Surveys*, 27-3, 1995.

Adi Ben-Israel and Thomas N.E. Greville. *Generalized Inverses: Theory and Applications*. Springer, second edition, 2003.

Gal Berkooz, Philip Holmes, and John L. Lumley. The Proper Orthogonal Decomposition in the Analysis of Turbulent Flows. *Annual Review of Fluid Mechanics*, 25:539–575, 1993.

Josef Bigün, Gö H. Granlund, and Johan Wiklund. Multidimensional orientation estimation with application to texture analysis and optical flow. *IEEE Journal of Pattern Analysis and Machine Intelligence*, 13(8):775–790, 1991.

M. Black, Y. Yacoob, A. Jepson, and D. Fleet. Learning Parameterized Models of Image Motion. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 1997.

Michael J. Black and Padmanabhan Anandan. Robust Dynamic Motion Estimation Over Time. In *Proc. Computer Vision and Pattern Recognition*, pages 296–302, Maui, Hawaii, 1991.

Michael J. Black and Padmanabhan Anandan. The Robust Estimation of Multiple Motions: Parametric and Piecewise-Smooth Flow Fields. *Computer Vision and Image Understanding*, 63:75–104, 1996.

Guido Boffetta. Energy and enstrophy fluxes in the double cascade of two-dimensional turbulence. *Journal of Fluid Mechanics*, 589:253–260, 2007.

Guido Boffetta, Antonio Cenedese, Stefania Espa, and Stefano Musacchio. Experimental study of two-dimensional enstrophy cascade. *Europhysics Letters*, 71(590), 2005.

William L. Briggs, Van Emden Henson, and Steve F. McCormick. *A multigrid tutorial*. SIAM, Philadelphia, second edition, 2000.

Thomas Brox. *From Pixels to Regions: Partial Differential Equations in Image Analysis*. PhD thesis, Saarland University, 2005.

Thomas Brox, Andrés Bruhn, Nils Papenberg, and Joachim Weickert. High Accuracy Optical Flow Estimation Based on a Theory for Warping. In *Proc. 8th European Conference on Computer Vision*, 2004.

Andrés Bruhn. *Variational Optic Flow Computation - Accurate Modelling and Efficient Numerics*. PhD thesis, Saarland University, 2006.

Andrés Bruhn, Joachim Weickert, and Christoph Schnörr. Lucas/Kanade Meets Horn/Schunck: Combining Local and Global Optic Flow Methods. *International Journal of Computer Vision*, 61(3):211–231, 2005.

Andrés Bruhn, Joachim Weickert, Timo Kohlberger, and Christoph Schnörr. A Multigrid Platform for Real-Time Motion Computation with Discontinuity-Preserving Variational Methods. *International Journal of Computer Vision*, 70(3):257–277, 2006. doi: 10.1007/s11263-006-6616-7.

Johan Carlier. 2d turbulence sequence. *Provided by CEMAGREV within the European project 'Fluid Image Analysis and Description' http://fluid.irisa.fr/*, 2005a.

Johan Carlier. Second set of fluid mechanics image sequences. *European Project 'Fluid image analysis and description' (FLUID)*, 2005b.

Cyril Cassisa, Serge Simoens, Véronique Prinet, and Liang Shao. Subgrid scale formulation of optical flow for the study of turbulent flow. *Experiments in Fluids*, 51:1739–1754, 2011.

Tony F. Chan and Pep Mulet. On the convergence of the lagged diffusivity fixed point method in total variation image restoration. *SIAM Journal on Numerical Analysis*, 36(2):354–367, 1999.

Anindya Chatterjee. An introduction to the proper orthogonal decomposition. *Current Science*, 78(7):808–817, 2000.

Nancy Cornelius and Takeo Kanade. Adapting Optical-Flow to Measure Object Motion in Reflectance and X-ray Image Sequences. Technical report, Carnegie Mellon University, Computer Science Department, 1983.

Thomas Corpetti, Étienne Mémin, and Patrick Pérez. Dense Estimation of Fluid Flows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:365–380, 2002.

Thomas Corpetti, Dominique Heitz, Georges Arroyo, Étienne Mémin, and Alina Santa-Cruz. Fluid experimental flow estimation based on an optical-flow scheme. *Experiments in Fluids*, 40:80–97, 2006.

Konstantinos G. Derpanis. Characterizing Image Motion. Technical report, York University, Ontario, Canada, 2006.

Gerrit E. Elsinga, Fulvio Scarano, Bernhard Wieneke, and Bas W. van Oudheusden. Tomographic particle image velocimetry. *Experiments in Fluids*, 41:933–947, 2006.

EMVA. EMVA Standard 1288 - Standard for Characterization of Image Sensors and Cameras. Technical Report Release 3.0, European Machine Vision Association, November 2010.

Richard Everson and Lawrence Sirovich. Karhunen-Loève procedure for gappy data. *Journal of the Optical Society of America A*, 12(8):1657–1646, 1995.

Marco Fahl. *Trust-Region Methods for Flow Control based on Reduced Order Modelling*. PhD thesis, University of Trier, 2000.

Joel H. Ferziger, Koseff Jeffrey R., and Stephen G. Monismith. Numerical simulation of geophysical turbulence. *Computers & Fluids*, 31:557–568, 2002.

David Fleet, Michael J. Black, Yaser Yacoob, and Allan Jepson. Design and Use of Linear Models for Image Motion Analysis. *International Journal of Computer Vision*, 36(3):171–193, 2000.

David J Fleet and Allan D. Jepson. Computation of Component Image Velocity from Local Phase Information. *International Journal of Computer Vision*, 5:1:77–104, 1990.

David J Fleet and Yair Weiss. *Handbook of Mathematical Models in Computer Vision*, chapter Optical Flow Estimation. Springer, 2006.

Frank Glazer, George Reynolds, and Padmanabhan Anandan. Scene Matching by Hierarchical Correlation. Technical report, Massachusetts University Amherst Department of Computer and Information Science, 1983.

Gene H. Golub and Charles F. Van Loan. *Matrix Computations, Third Edition*. Johns Hopkins University Press, 1996.

Stanislav V. Gordeyev and Flint O. Thomas. Coherent structure in the turbulent planar jet. Part 1. Extraction of proper orthogonal decomposition eigenmodes and their self-similarity. *Journal of Fluid Mechanics*, 414:145–194, 2000.

Jacques Hadamard. Sur les problèmes aux dérivés partielles et leur signification physique. *Princeton University Bulletin*, 13:49–52, 1902.

George Haller. An objective definition of a vortex. *Journal of Fluid Mechanics*, 525:1–26, 2005. doi: 10.1017/S0022112004002526.

George Haller and Guocheng Yuan. Lagrangian coherent structures and mixing in two-dimensional turbulence. *Physica D*, 147:352–370, 2000.

Horst Haussecker and Hagen Spies. *Handbook of Computer Vision and Applications*, chapter 13, pages 309–396. Academic Press, 1999.

Horst W. Haussecker and David J. Fleet. Computing Optical Flow with Physical Models of Brightness Variation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23: 661–673, 2001.

Patrick Héas, Cédric Herzet, Etienne Mémin, Dominique Heitz, and Pablo D. Mininni. Bayesian Estimation of Turbulent Motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(6):1343–1356, 2013.

David J. Heeger. Optical flow using spatiotemporal filters. *International Journal of Computer Vision*, 1:279–302, 1988.

Dominique Heitz, Etienne Mémin, and Christoph Schnörr. Variational fluid flow measurements from image sequences: synopsis and perspectives. *Experiments in Fluids*, 48:369–393, 2010.

Klaus D. Hinsch. Holographic particle image velocimetry. *Measurement Science and Technology*, 13:R61–R72, 2002.

Philip Holmes, John L. Lumley, and Gal Berkooz. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge Monographs on Mechanics, 1996.

Berthold K. P. Horn. *Robot Vision*. MIT Press, 1986.

Berthold K. P. Horn and Brian G. Schunck. Determining Optical Flow. *Artificial Intelligence*, 17:185–203, 1981.

Peter J. Huber. *Robust Statistics*. John Wiley & Sons, 1981.

J.C.R. Hunt, A.A. Wray, and P. Moin. Eddies, Streams, and Convergence Zones in Turbulent Flows. In *Center for Turbulence Research, Proceedings of the Summer Program*, 1988.

Bernd Jähne. *Digital Image Processing*. Springer, 2005.

Bernd Jähne, Michael Klar, and Markus Jehle. *Handbook of Experimental Fluid Mechanics*, chapter 25, pages 1437–1491. Springer, 2007.

Richard D. Keane and Roland J. Adrian. Optimization of particle image velocimeters. Part I: Double pulsed systems. *Measurement Science and Technology*, 1:1202–1215, 1990.

Richard D. Keane and Roland J. Adrian. Theory of cross-correlation analysis of PIV images. *Applied Scientific Research*, 49:191–215, 1992.

Shigeo Kida and Hideaki Miura. Identification and analysis of vortical structures. *European Journal of Mechanics - B/Fluids*, 17(4):471–488, 1998.

Claudia Kondermann, Daniel Kondermann, Bernd Jähne, and Christoph Garbe. An Adaptive Confidence Measure for Optical Flows Based on Linear Subspace Projections. In *Pattern Recognition*, volume 4713 of LNCS. Springer, 2007.

Pijush K. Kundu, Ira M. Cohen, and David R. Dowling. *Fluid Mechanics*. Academic Press, fifth edition, 2011.

LaVision. Data aquisition and Visualisation (DaVis), 2014. URL `http://www.lavision.de/de/products/davis.php`. Accessed: 2014-10-31.

Leonid P. Lebedev and Michael J. Cloud. *The Calculus of Variations and Functional Analysis*. World Scientific, 2003.

Stan Z. Li and Anil K. Jain. *Handbook of Face Recognition*. Springer, 2nd edition, 2011.

Ce Liu. *Beyond Pixels: Exploring New Representations and Applications for Motion Analysis*. PhD thesis, Massachusetts Institute of Technology, 2009.

Tianshu Liu and Lixin Shen. Fluid flow and optical flow. *Journal of Fluid Mechanics*, 614: 253–291, 2008.

Bruce D. Lucas and Takeo Kanade. An Iterative Image Registration Technique with an Application to Stereo Vision (DARPA). *Proceedings of the 1981 DARPA Image Understanding Workshop*, pages 121–130, 1981.

John L. Lumley. The structure of inhomogeneous turbulent flows. In *Atmospheric Turbulence and Radio Wave Propagation*, 1967.

John L. Lumley. Coherent structures in turbulence. In *Transition and turbulence*, 1981.

Etienne Mémin and Patrick Pérez. Dense Estimation and Object-Based Segmentation of the Optical Flow with Robust Techniques. *IEEE Transactions on Image Processing*, 7:703–719, 1998.

Etienne Mémin and Patrick Pérez. Hierarchical estimation and segmentation of dense motion fields. *International Journal of Computer Vision*, 46(2):129–155, 2002.

Wolfgang Merzkirch. *Handbook of Experimental Fluid Mechanics*, chapter 11, pages 857–870. Springer, 2007.

Hans-Hellmut Nagel. On a Constraint Equation for the Estimation of Displacement Rates in Image Sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(1): 13–30, 1989.

Hans-Hellmut Nagel and Wilfried Enkelmann. An Investigation of Smoothness Constraints for the Estimation of Displacement Vector Field from Image Sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Pami 8(5):565–593, 1986.

Hans-Hellmut Nagel and A. Gehrke. Spatitemporal adaptive estimation and segmentation of OF-fields. In *Proceedings of the ECCV, Lecture Notes in Computer Science*, pages 87–102. Springer, 1998.

Yoshikazu Nakajima, Hiroshi Inomata, Hiroki Nogawa, Yoshinobu Sato, Shinichi Tamura, Kozo Okazaki, and Seiji Torii. Physics-based flow estimation of fluids. *Pattern Recognition*, 36: 1203–1212, 2003.

Shahriar Negahdaripour and Chih-Ho Yu. A Generalized Brightness Change Model for Computing Optical Flow. In *IEEE International Conference on Computer Vision*, 1993.

Claudia Nieuwenhuis. *Postprocessing and Restoration of Optical Flows.* PhD thesis, Ruprecht Karls University Heidelberg, 2009.

Claudia Nieuwenhuis, Daniel Kondermann, and Christoph Garbe. Complex Motion Models for Simple Optical Flow Estimation. *Pattern Recognition (Proc. DAGM) LNCS*, 6376:141–150, 2010.

Holger Nobach, Cameron Tropea, Laurent Cardier, Jean-Paul Bonnet, Joël Delville, Jacques Lewalle, Marie Farge, Kai Schneider, and Ronald J. Adrian. *Handbook of Experimental Fluid Mechanics*, chapter 22, pages 1337–1398. Springer, 2007.

PIV-Challenge. Homepage of the PIV Challenge, 2014. URL `http://www.pivchallenge.org/`. Accessed: 2014-10-31.

Ludwig Prandtl. Über Flüssigkeitsbewegung bei sehr kleiner Reibung. In *Verhandlungen des III. Internationalen Mathematiker-Kongresses, Heidelberg*, 1904.

Ludwig Prandtl and Albert Betz. *Vier Abhandlungen zur Hydrodynamik und Aerodynamik, Göttinger Klassiker der Strömungsmechanik Bd. 3*. Universitätsverlag Göttingen, 2010.

Alfio Quarteroni, Riccardo Sacco, and Fausto Saleri. *Numerical Mathematics*. Springer, 2000.

Georges M. Quénot, Jaroslaw Pakleza, and Tomasz A. Kowalewski. Particle image velocimetry with optical flow. *Experiments in Fluids*, 25:177–189, 1998.

Samuel G. Raben, John J. Charonko, and Pavlos P. Vlachos. Adaptive gappy proper orthogonal decomposition for particle image velocimetry data reconstruction. *Measurement Science and Technology*, 23(2), 2012.

Markus Raffel, Christian E. Willert, Steven T. Wereley, and Jürgen Kompenhans. *Particle Image Velocimetry: A practical guide*. Springer Verlag, 2007.

Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

Paul Ruhnau and Christoph Schnörr. Optical stokes flow estimation: an imaging-based control approach. *Experiments in Fluids*, 42:61–78, 2007.

Paul Ruhnau, Timo Kohlberger, Christoph Schnörr, and Holger Nobach. Variational optical flow estimation for particle image velocimetry. *Experiments in Fluids*, 38:21–32, 2005.

Paul Ruhnau, Anette Stahl, and Christoph Schnörr. Variational estimation of experimental fluid flows with physics-based spatio-temporal regularization. *Measurement Science and Technology*, 18:755–763, 2007.

Fulvio Scarano. Iterative image deformation methods in PIV. *Measurement Science and Technology*, 13:R1–R19, 2002.

Fulvio Scarano and Michel L. Riethmuller. Iterative multigrid approach in PIV image processing with discrete window offset. *Experiments in Fluids*, 26:513–523, 1999.

Fulvio Scarano and Michel L. Riethmuller. Advances in iterative multigrid PIV image processing. *Experiments in Fluids*, Suppl.:S51–S60, 2000.

Hanno Scharr. *Optimal Operators in Digital Image Processing*. PhD thesis, Heidelberg University, 2000.

Hanno Scharr. Optimal Filters for Extended Optical Flow. In *Complex Motion, Lecture Notes in Computer Science. Volume 3417*, 2007.

Brian G. Schunck. The motion constraint equation for optical flow. In *Proceedings International Conference on Pattern Recognition*, pages 20–22, 1984.

Brian G. Schunck. Image flow: Fundamentals and future research. In *IEEE Conference on Computer Vision and Pattern Recognition, San Francisco*, pages 560–571, 1985.

Hans Rudolf Schwarz and Norbert Köckler. *Numerische Mathematik*. Vieweg + Teubner, 8. edition, 2011.

Eero P. Simoncelli, Edward H. Adelson, and David J. Heeger. Probability distributions of optical flow. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 310–315, 1991.

Lawrence Sirovich. Turbulence and the Dynamics of Coherent Structures. Part I: Coherent Structures. *Quarterly of Applied Mathematics*, 45(3):561–571, 1987.

Joseph H. Spurk and Aksel Nuri. *Fluid Mechanics*. Springer, second edition, 2008.

Michel Stanislas, Koji Okamoto, Christian J Käler, Jerry Westerweel, and Fulvio Scarano. Main results of the third international PIV Challenge. *Experiments in Fluids*, 45:27–71, 2008.

Julian Stapf and Christoph Garbe. Partikelbasierte Strömungsmessung unter Ausnutzung typischer Bewegungsmuster. In *Lasermethoden in der Strömungsmesstechnik*, München, September 2013. Deutsche Gesellschaft für Laser-Anemometrie GALA e.V.

Julian Stapf and Christoph Garbe. A learning-based approach for highly accurate measurements of turbulent fluid flows. *Experiments in Fluids*, 55(8), 2014a. doi: 10.1007/s00348-014-1799-0.

Julian Stapf and Christoph Garbe. A variational optical flow approach using learned motion models for the determination of fluid flows. In *17th International Symposium on Applications of Laser Techniques to Fluid Mechanics*, Lisbon, Portugal, July 2014b.

Charles V. Stewart. Robust Parameter Estimation in Computer Vision. *SIAM Rieview*, 41(3): 513–537, 1999.

R. Strzodka and C.S. Garbe. Real-Time Motion Estimation and Visualization on Graphics Cards. In *Proceedings IEEE Visualization 2004*, pages 545–552, 2004.

Deqing Sun, Stefan Roth, and Michael J. Black. A Quantitative Analysis of Current Practices in Optical Flow Estimation and the Principles Behind Them. *International Journal of Computer Vision*, 106:115–137, 2014.

David Suter. Motion Estimation and Vector Splines. In *Proceedings Conference on Computer Vision and Pattern Recognition*, pages 939–942, 1994.

Richard Szeliski. Image alignment and stitching: A tutorial. *Foundations and Trends in Computer Graphics and Computer Vision*, 2(1):1–104, 2006.

Richard Szeliski. *Computer Vision*. Springer, 2011.

Andrey N. Tikhonov and Vasily Y. Arsenin. *Solutions of Ill Posed Problems*. John Wiley & Sons Inc., 1977.

O. Tretiak and L. Pastor. Velocity estimation from image sequences with second order differential operators. In *IEEE International Conference on Pattern Recognition*, pages 16–19, 1984.

Matthew Turk and Alex Pentland. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

Shimon Ullman. *The Interpretation of Visual Motion*. MIT Press, Cambridge, MA, 1979.

Sabine Van Huffel and Joos Vandewalle. *The Total Least Squares Problem: Computational Aspects and Analysis*. Society for Industrial and Applied Mathematics, 1991.

Alessandro Verri and Poggio Tomaso. Motion Field and Optical Flow: Qualitative Properties. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(5):490–498, 1989.

Alessandro Verri, Federico Girosi, and Vincent Torre. Differential techniques for optical flow. *Journal of the Optical Society of America A*, 7(5):912–922, 1990.

Stefan Volkwein. Proper Orthogonal Decomposition and Singular Value Decomposition. Technical report, Institut für Mathematik, Universität Graz, 1999.

Allen Waxmann, J. Wu, and Bergholm Fredrik. Convected activation profiles and receptive fields for real time measurement of short range visual motion. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 1988.

Joachim Weickert and Christoph Schnörr. A Theoretical Framework for Convex Regularizers in PDE-Based Computation of Image Motion. *International Journal of Computer Vision*, 45 (3):245–264, 2001.

Tom Weier, Thomas Albrecht, Gunter Gerbeth, Sebastian Wittwer, Hans Metzkes, and J"org Stiller. The Electromagnetically Forced Flow over a Backward-Facing Step. In *TSFP7 Ottawa*, 2011.

Richard P. Wildes, Michael J. Amabile, Ann-Marie Lanzillotto, and Tzong-Shyng Leu. Physically based fluid flow recovery from image sequences. In *Proceedings Conference on Computer Vision and Pattern Recognition*, pages 969–975, 1997.

Christian E. Willert and M Gharib. Digital particle image velocimetry. *Experiments in Fluids*, 10:181–193, 1991.

Yaser Yacoob and Larry Davis. Learned Temporal Models of Image Motion. In *International Conference on Computer Vision*, 1998.

Jing Yuan, Christoph Schnörr, and Étienne Mémin. Discrete Orthogonal Decomposition and Variational Fluid Flow Estimation. *Journal of Mathematical Imaging and Vision*, 28:67–80, 2007.

Danksagung

Ich möchte allen danken, die zum Gelingen dieser Arbeit beigetragen haben. Besonderen Dank gilt meinem Betreuer Priv.-Doz. Dr. Christoph Garbe, der mir ermöglichte in seiner Arbeitsgruppe *Image Processing and Modeling* zu promovieren. Mit vielen Ideen und Tipps stand er mir stets tatkräftig zur Seite. Bei Prof. Dr. Werner Aeschbach-Hertig bedanke ich mich, dass er sich die Zeit genommen hat, die Arbeit zu begutachten. Prof. Dr. Dirk Dubbers und Prof. Dr. Tilman Plehn danke ich für die Bereitschaft Teil meiner Prüfungskommission zu sein.

Für die finanzielle Unterstützung dieser Dissertation durch ein Stipendium der Deutschen Forschungsgemeinschaft im Rahmen des Graduiertenkollegs GRK1114 *Optische Messtechniken für die Charakterisierung von Transportprozessen an Grenzflächen* der TU Darmstadt und der Universität Heidelberg möchte ich mich bei den Trägern des Graduiertenkollegs bedanken. Ebenso möchte ich mich bei der Heidelberger Graduiertenschule *HGS MathComp* bedanken, die meine Teilnahme an verschiedenen Konferenzen finanzierte.

Bei allen ehemaligen und aktuellen Kollegen am HCI, dem IUP, dem GRK und der HGS bedanke ich mich für die schöne gemeinsame Zeit und die tolle Zusammenarbeit.

Für die Wartung und Betreuung der Server sowie die Unterstützung bei Rechnerproblemen aller Art bedanke ich mich bei Dominic Spangenberger, Dr. Nils Krah sowie Jürgen Moldenhauer. Bei Sascha Hub möchte ich mich für die Hilfe beim Wiederherstellen einiger Dateien meiner zerstörten Festplatte bedanken.

Des Weiteren geht mein Dank an Karin Kruljac und Dr. Christian Schmidt für die Unterstützung bei administrativen Fragen.

Für das Korrekturlesen dieser Arbeit bedanke ich mich ganz herzlich bei Prof. Dr. Hiltrud Brauch, Jana Schnieders, Marcel Gutsche, Dr. Felix Friedl, Dr. Alfred Klar, Florian Stapf, Günter Stapf sowie Corinna Korder.